(2)

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

INSTANTANEOUS POWER SPECTRUM AND 1½D INSTANTANEOUS POWER SPECTRUM TECHNIQUES

by

Karen Allyn Hagerman

June 1992

Co-Advisor: R. Hippenstiel
Co-Advisor: M.P. Fargues

Approved for public release; distribution is unlimited

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No 0704-0188

1a REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b RESTRICTIVE MARKINGS
2a SECURITY CLASSIFICATION AUTHORITY	3 DISTRIBUTION / AVAILABILITY OF REPORT Approved for public release; distribution is unlimited
2b DECLASSIFICATION / DOWNGRADING SCHEDULE	
4 PERFORMING ORGANIZATION REPORT NUMBER(S)	5 MONITORING ORGANIZATION REPORT NUMBER(S)
6a NAME OF PERFORMING ORGANIZATION Naval Postgraduate School 6b OFFICE SYMBOL (If applicable) EC	7a NAME OF MONITORING ORGANIZATION Naval Postgraduate School
6c ADDRESS (City, State, and ZIP Code) Monterey, CA 93943-5000	7b ADDRESS (City, State, and ZIP Code) Monterey, CA 93943-5000
8a NAME OF FUNDING / SPONSORING ORGANIZATION 8b OFFICE SYMBOL (If applicable)	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c ADDRESS (City, State, and ZIP Code)	10 SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO / PROJECT NO / TASK NO / WORK UNIT ACCESSION NO

11 TITLE (Include Security Classification) INSTANTANEOUS POWER SPECTRUM
INSTANTANEOUS POWER SPECTRUM TECHNIQUES

12 PERSONAL AUTHOR(S)
HAGERMAN, Karen Allyn

13a TYPE OF REPORT	13b TIME COVERED	14 DATE OF REPORT (Year, Month, Day)	15 PAGE COUNT
Master's Thesis	FROM _____ TO _____	1992 June	69

16 SUPPLEMENTARY NOTATION The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the US Government.

17 COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	IPS; 1½D; cumulant; higher order spectrum

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

The IPS (Instantaneous Power Spectrum) spectral analyis technique has been the subject of study for many years. This thesis implemented the IPS algorithm using MATLAB. In addition, two additional programs were written to deal with progressively larger data sets. Based on a third order cumulant, the 1 ½D spectral analysis technique, thought to perform well in low signal to noise environments, is also explored.

20 DISTRIBUTION / AVAILABILITY OF ABSTRACT	21 ABSTRACT SECURITY CLASSIFICATION
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	UNCLASSIFIED
22a NAME OF RESPONSIBLE INDIVIDUAL HIPPENSTIEL R./FARGUES, M.	22b TELEPHONE (Include Area Code) 408-646-2633/2859 22c OFFICE SYMBOL EC/Hi EC/Fa

DD Form 1473, JUN 86 *Previous editions are obsolete* SECURITY CLASSIFICATION OF THIS PAGE

S/N 0102-LF-014-6603

UNCLASSIFIED

Approved for public release; distribution is unlimited

Instantaneous Power Spectrum in 1½D Instantaneous Power Spectrum Techniques

by

Karen Allyn Hagerman
Lieutenant, United States Navy
B.A., Roosevelt University, 1984

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
June 1992

Author: Karen Allyn Hagerman
Karen Allyn Hagerman

Approved by: Ralph Hippenstiel
Ralph Hippenstiel, Co-Advisor

Monique Fargues, Co-Advisor

Michael A. Morgan
Michael A. Morgan, Chairman
Department of Electrical and Computer Engineering

ABSTRACT

The IPS (Instantaneous Power Spectrum) spectral analysis technique has been the subject of study for many years. This thesis implemented the IPS algorithm using MATLAB. In addition, two additional programs were written to deal with progressively larger data sets. Based on a third order cumulant, the 1 ½ D spectral analysis technique, thought to perform well in low signal to noise environments, is also explored.

TABLE OF CONTENTS

ACKNOWLEDGMENTS

To Patrick for his infinite patience and support.

To David for his help in "all the little things."

To Dori and Karen for their understanding through the years.

To Professor Hippenstiel and Professor Fargues for their constant availability and instruction.

I. STATIONARY SPECTRAL TECHNIQUES

A. INTRODUCTION

In the disparate fields of geology, communications, astronomy, oceanography, chemistry and biomedicine, observations of physical processes are typically collected and then analyzed to extract as much information as possible. These observations are primarily analog and continuous in time. Spectral analysis is one of the initial analysis tools available; it is used to determine what frequency components are present in the observation. This analysis includes information about the spectral magnitude of the frequency components. Many spectral analysis techniques have been developed to investigate this problem. The first scientific application of spectral analysis, was in chemistry and astronomy. Light, whether from an astral object or from the combustion of a chemical sample was split into its component parts by prisms and the resulting spectra were photographed for later analysis [Ref. 1]. With the development of instruments which could record electromagnetic, seismic or acoustic information, additional spectral analysis techniques were developed to analyze these recorded observations.

Classical spectral analysis techniques are based on Fourier methods and include the Periodogram [Ref. 2] and its time-frequency analog, the Spectrogram [Ref. 4], both of which will be discussed further in this chapter. The Fourier based methods are particularly well suited as computer-based analysis tools. The continuous-time data sequence is appropriately filtered and sampled to produce discrete data elements which can then be easily processed with the Fast Fourier Transform (FFT), a fast version of the Discrete Fourier Transform (DFT).

Spectral analysis techniques based on linear filter models were also developed [Ref. 3]. The goal of these methods is to design a linear filter, $H(e^{j\omega})$, which when driven by stationary white noise can produce the data in a statistical sense. As an example, the Autoregressive (AR) model will return an all-pole filter with a resultant spectrum defined as

$$\hat{S}_{AR}(e^{j\omega}) = \frac{|b_o|^2}{|A(e^{j\omega})|^2}; \qquad (1.1)$$

where $A(e^{j\omega})$ is the Fourier transform of the pole coefficients and b_o represents the DC gain of the filter. The AR model is well-suited to the detection of discrete sinusoidal signals or other narrow band signals. Other model-based spectral estimation techniques include both the Moving Average (MA) and the Autoregressive-Moving-Average (ARMA) models [Ref. 3]. Each of these has advantages and disadvantages depending on assumptions that can be made concerning the data sequence. All of these model-based methods, however, assume that the data sequence is Wide Sense Stationary (WSS). Many data sequences, however, are not WSS that is, the signals of interest are transient in a time domain sense or have dynamic spectral features. In these cases, the AR, ARMA and MA spectral techniques will not be adequate.

Other spectral estimation techniques are based on eigenvalue decomposition or subspace methods [Ref. 7]. These methods presuppose that the data can be separated into a noise subspace and a signal subspace. The object, then, is to correctly identify which eigenvalues belong to the noise subspace and which belong to the signal subspace. Spectral techniques based on subspaces include Pisarenko, MUSIC and ESPRIT techniques [Ref. 3]. These techniques do not actually provide a classical spectral display, their spectra consists of delta functions representing the sinusoidal components. One

major weakness of these methods is the fact that they were designed to isolate narrow band signals and are less suited for wide band signals.

This thesis will explore two other spectral estimation techniques which are based on instantaneous estimates of the second and third moments. The first of these techniques to be explored is the Instantaneous Power Spectrum (IPS) [Ref. 4] which is based on an estimate of the second moment. It differs from the Periodogram, which is also based on an estimate of the second moment, by using an instantaneous estimate of the autocorrelation sequence. IPS belongs to a class of distributions called "Cohen's class" [Ref. 3]. The other technique to be explored is the 1 ½ D Instantaneous Power Spectrum (1 ½D) [Ref. 6], which is based on third moment properties of the data.

B. CLASSICAL SPECTRAL ESTIMATION TECHNIQUES

All spectral estimation techniques are concerned with determining the spectral content of a finite set of observations. The formal definition of the *Power spectral density* (PSD) [Ref. 3] of a Wide Sense Stationary (WSS) discrete time random process is

$$P_{xx}(e^{j\omega}) = \sum_{k=-\infty}^{\infty} r_{xx}(k)e^{-j\omega k} \quad -\pi < \omega \leq \pi; \tag{1.2}$$

where $r_{xx}(k)$ is the autocorrelation function of the observation, x(n), defined as

$$r_{xx}(k) = \mathrm{E}\left[x^*(n)x(n+k)\right] \tag{1.3}$$

and where **E** denotes the expectation operator. To obtain the expected value, time averaging is usually used requiring essentially the observation of an ergodic process over an infinite period. The PSD displays the frequency components of any WSS random process of infinite duration. However, data observations are always of finite duration and wide-sense-stationary only in a local sense. The challenge then, is to approximate the true

PSD with as much fidelity as possible. Classical spectral estimation techniques include the Periodogram and its time-frequency analog, the Spectrogram.

1. Periodogram

The Periodogram is defined as

$$\hat{P}_x(e^{j\omega}) = \left|X(e^{j\omega})\right|^2 \tag{1.4}$$

where $\hat{P}_x(e^{j\omega})$ is the estimated PSD and $X(e^{j\omega})$ is the Fourier transform of the observation *x(n)* of length N_x. One of the inherent weaknesses of the Periodogram is the fact that it deals with a finite set of observations. A finite set of observations can be obtained from an infinite set by applying a rectangular window of the form

$$W_R(k) = \begin{cases} 1; & 1 \le k \le N_x \\ 0; & \text{otherwise} \end{cases} \tag{1.5}$$

This time domain operation appears as a periodic convolution in the frequency domain. That is, the Fourier transform of the rectangular window is convolved with the Fourier transform of the observation set. The Fourier transform of the rectangular window is the digital ***sinc*** function which tends to smear the true PSD. To illustrate this point we can create an analytic sinusoid at some arbitrary frequency whose true Fourier transform would be an impulse function at the appropriate frequency location. For Figures 1.1 through 1.3, an analytic signal in additive white Gaussian noise with a signal to noise ratio (SNR) of -10 dB was created. The signal to noise ratio is defined as

$$\text{SNR} = 10 * \log_{10}\left(\frac{Power_{signal}}{Power_{noise}}\right) \tag{1.6}$$

Periodograms were then taken with the following parameters:

- Figure 1.1, 64 point observation sequence using a 64 point Fourier Transform
- Figure 1.2, 64 point observation sequence using a 128 point Fourier Transform
- Figure 1.3, 64 point observation sequence using a 512 point Fourier Transform

Each Periodogram was plotted with a discrete plot function to enhance the display of the ***sinc*** function. In each figure the true frequency location of 15.78 is indicated by a straight line at the appropriate location.

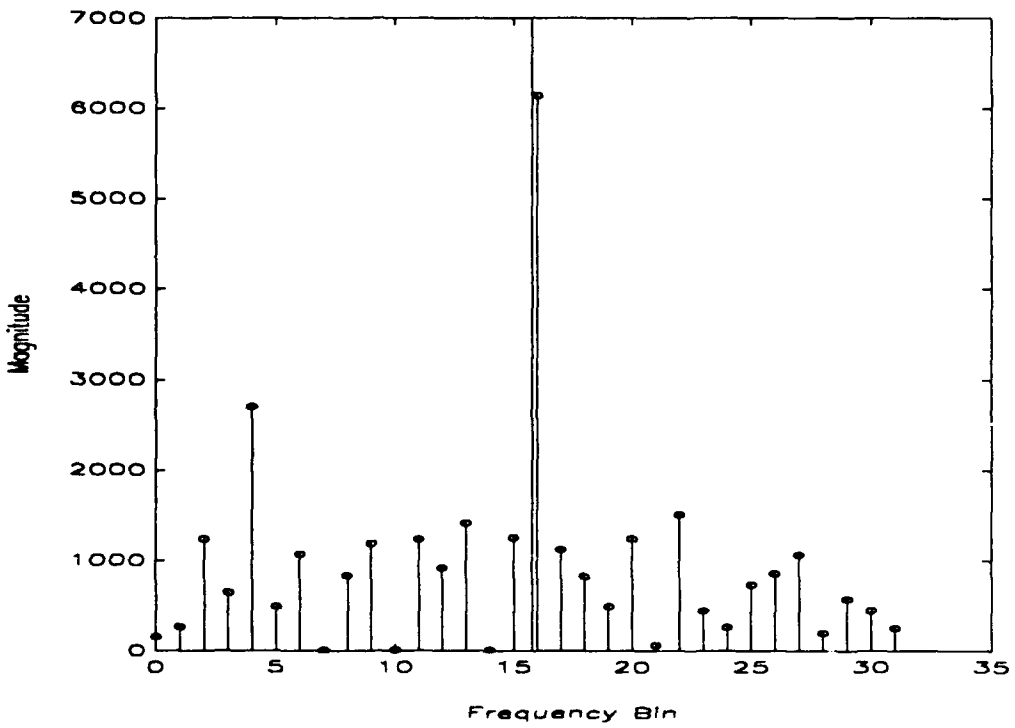


Figure 1.1. 64 point Periodogram

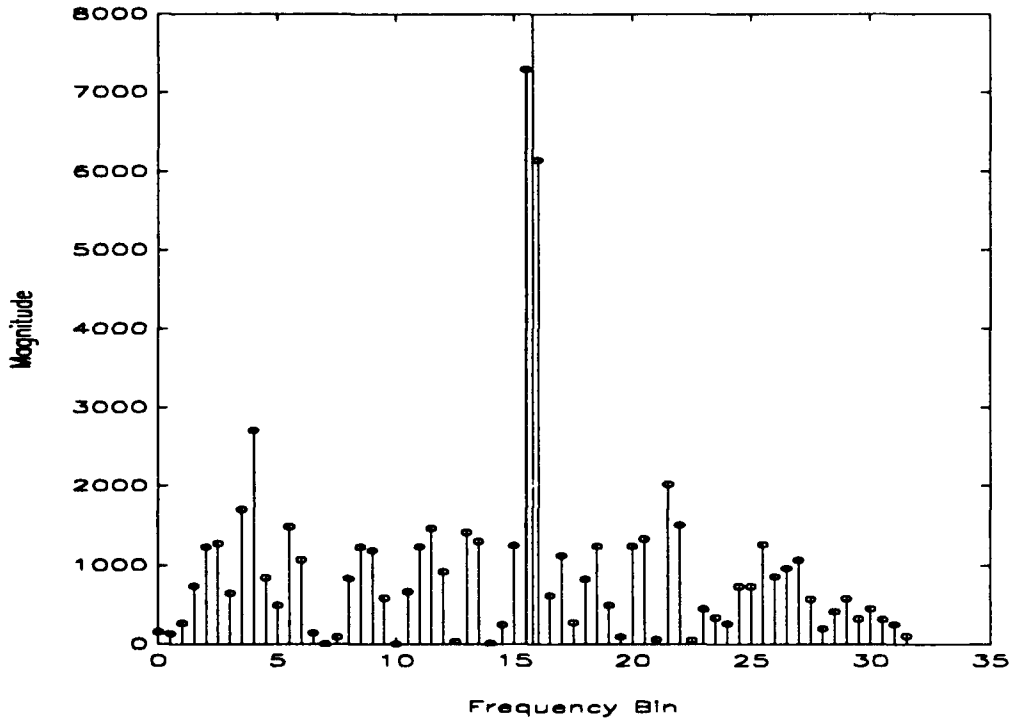


Figure 1.2. 128 point Periodogram

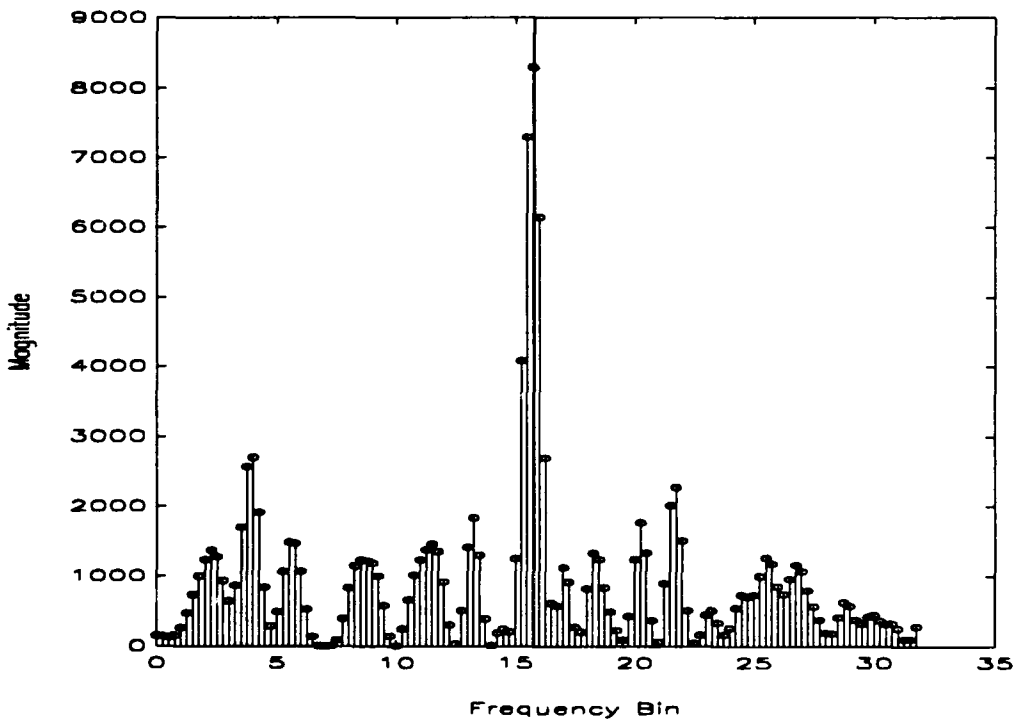


Figure 1.3. 512 point Periodogram

A comparison of Figures 1.1, 1.2 and 1.3 shows that as the transform length is increased, hence the number of frequency bins is increased, the position of the dominant spectral peak approaches more closely the true location of the frequency of the analytic signal. It can also be noted from these figures that although the apparent resolution improved as the transform length increased, the variance of the additive Gaussian white noise stayed at the same level. For the results in Figures 1.2 and 1.3, zeros were padded to the data sequence to obtain the transform lengths of 128 and 512, respectively. The variance of the spectral estimate is independent of the length of the data sequence. Typically, sequential periodograms are taken and then averaged to reduce the undesirable effects of the variance of the estimate [Ref. 2]. Figure 1.4 is a 512 point Fourier transform Periodogram of a noise-free data sequence created with two analytic sinusoids separated by just $\frac{1}{128}$th of the sampling frequency. As in the other figures, the true frequency locations are indicated by straight lines. It can be seen that the Periodogram correctly resolves the two narrow-band components, but even in this ideal, noise-free environment, the position of each spectral peak is slightly off the true frequency bin locations.

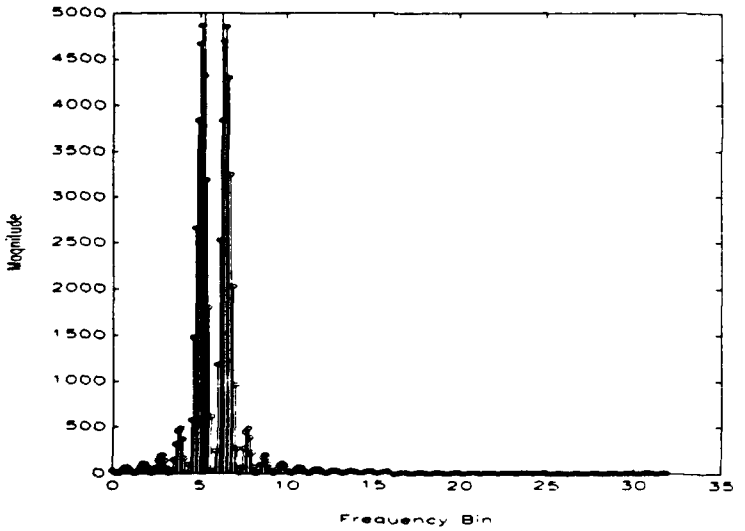


Figure 1.4. 512 point Fourier Transform Periodogram of Two Analytic Sinusoids in additive Gaussian White Noise at a SNR of -10 dB

Another weakness of the Periodogram is its inability to provide any information on the occurrence in time of an observation. To illustrate this an FSK data set composed of an analytic sinusoid which is switched from one frequency to another frequency at time 64 was created and is shown in Figure 1.5.

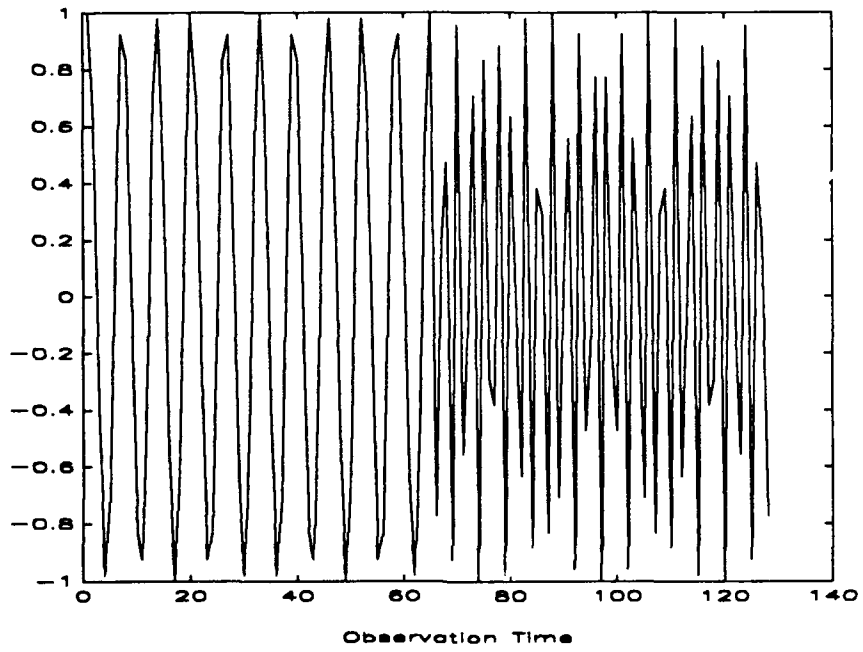


Figure 1.5. Time plot of Frequency Shift Key (FSK) signal

When the Periodogram (shown in Figure 1.6) is calculated for this FSK observation, the presence of both sinusoids is evident but the fact that they existed during different times in the observation set cannot be seen.

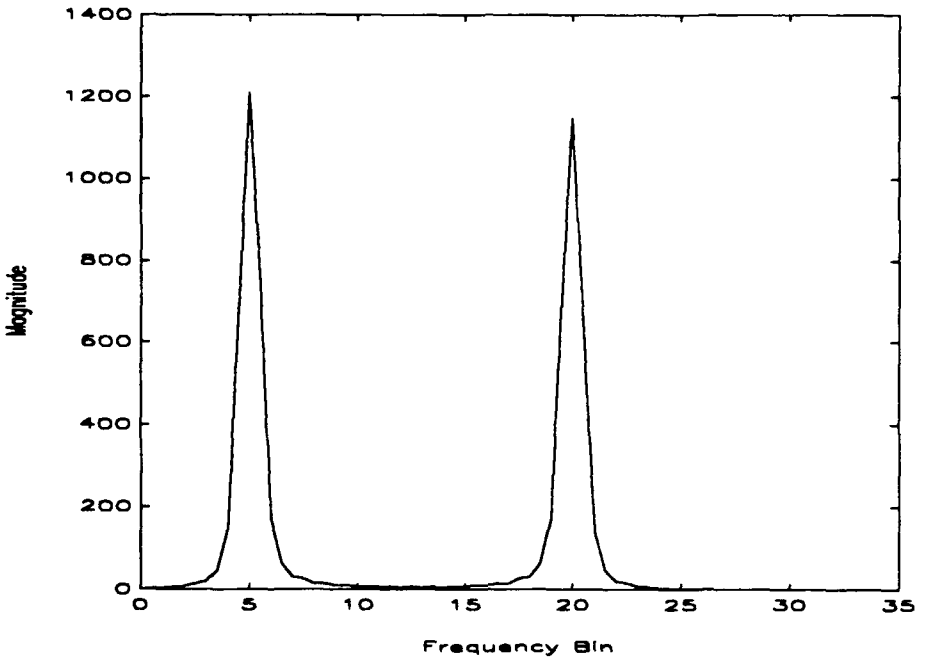
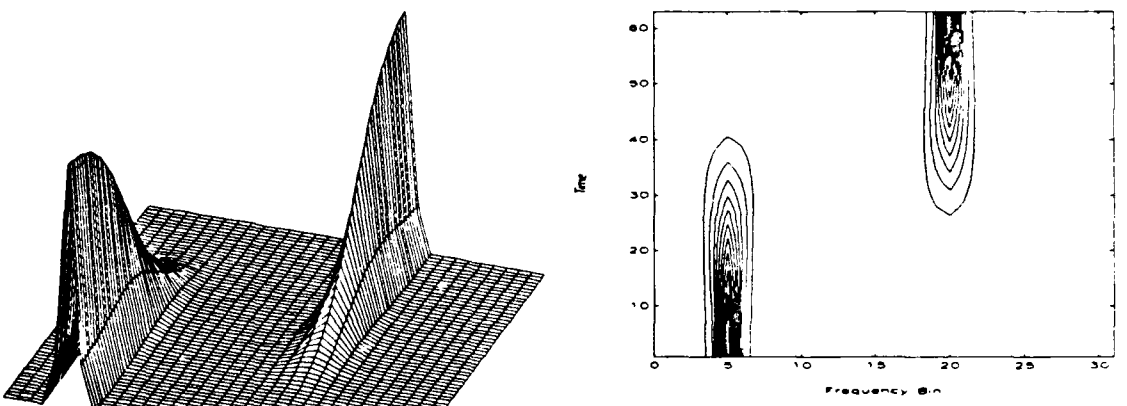


Figure 1.6. 128 point Fourier Transform Periodogram of a FSK Signal

The time-frequency representation of the Periodogram, the Spectrogram, can show that these signals were separated in time.

2. Spectrogram

The Spectrogram operates by applying a window to a subset of the data set and computing and saving the Periodogram of that subset. Then the window is stepped through the data by some fixed numbers of points and a Periodogram is recomputed. This process of stepping through the data and sequential computation of periodograms is repeated until the desired number of spectrogram lines is obtained. Figure 1.7 is a mesh and contour plot of the spectrogram of the FSK data set of Figure 1.5.



(a) (b)

Figure 1.7. Mesh and Contour plots of the Spectrogram of a FSK Signal

It can be seen that the frequency location is most concentrated at the beginning and end of the surface. At these points, the window includes a data segment which contains only one of the signals. As the window is stepped through the data set, the number of data points that include one frequency or the other changed. The Spectrogram surface seems to indicate that the first frequency at frequency location 5 is present from time 0 to time 41 when in fact it was only present from time 0 to 32. On the other hand, the second signal at frequency location 20 begins at time 32 not at time 26 which seems to be indicated.

An additional test signal whose frequency changes linearly with time, such as the linear FM Chirp, was created. Figure 1.8 is the Periodogram of a linear FM Chirp.

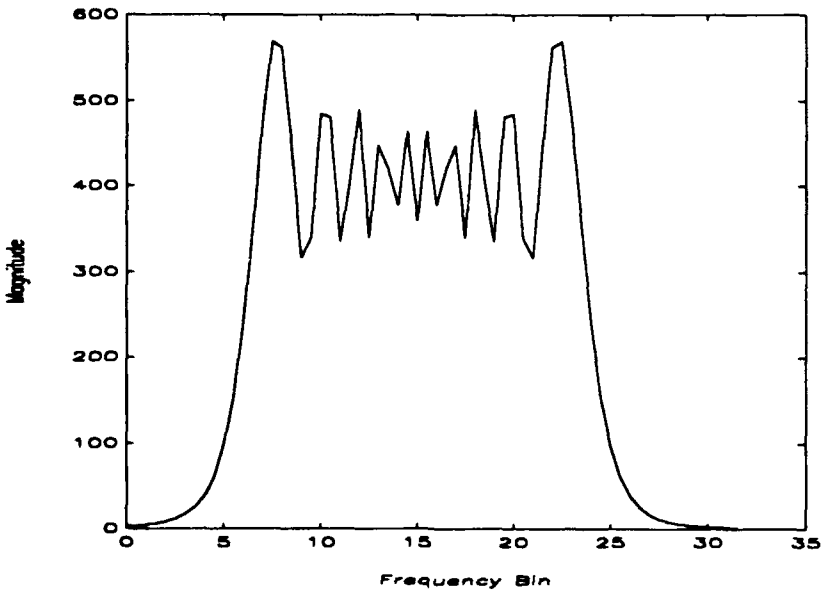
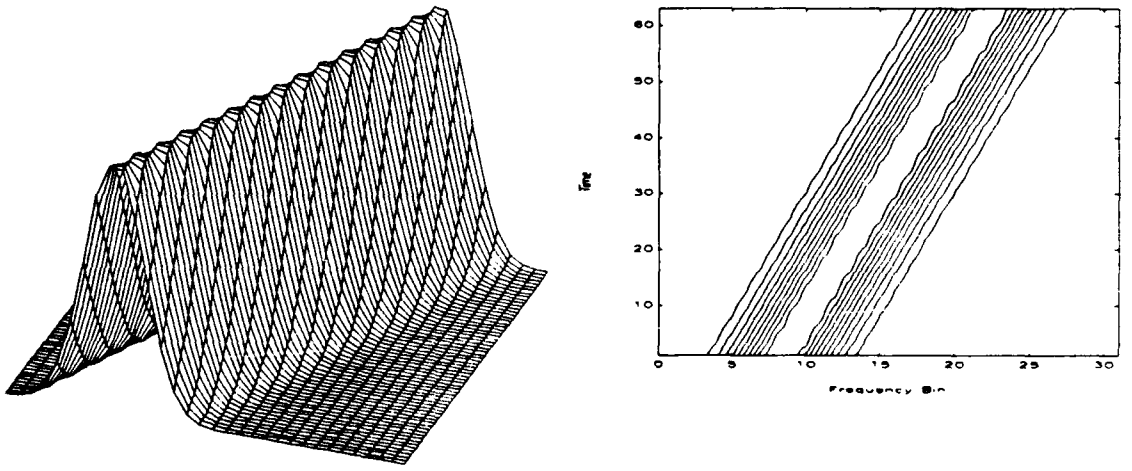


Figure 1.8. Periodogram of a Linear FM Chirp

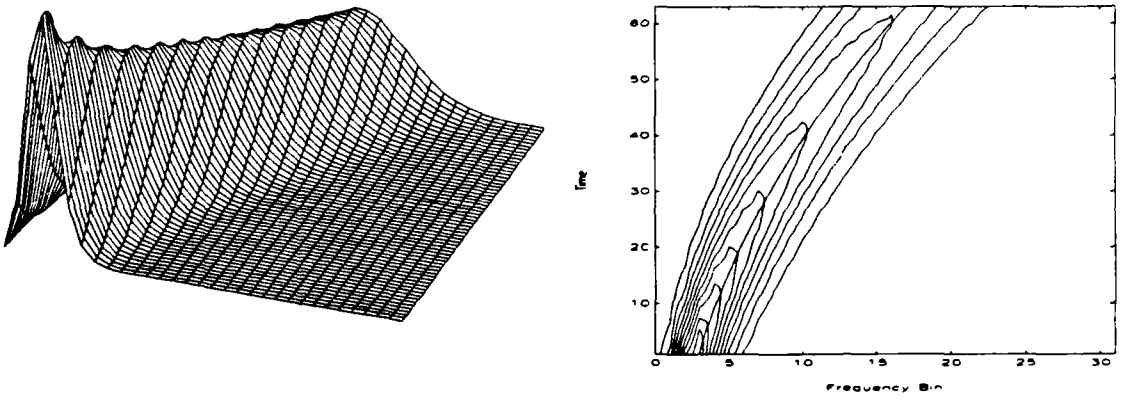
The Periodogram of Figure 1.8 indicates the presence of several sinusoidal components rather than the presence of a linear FM chirp. The Spectrogram in Figure 1.9, however, clearly shows that the data is a linear FM chirp.



(a) (b)

Figure 1.9. Spectrogram of a Linear FM Chirp

Using the linear FM chirp as an example of a dynamic signal, it can be seen that the width of the spectral lobe is wider than the spectral lobes for the FSK signal in Figure 1.7. To further illustrate this phenomenon a data set composed of a signal whose frequency changes as a quadratic function of time, a quadratic FM chirp, from frequency location 1 to location 30 was created. Figure 1.10 is the spectrogram of this signal.



(a) (b)

Figure 1.10. Spectrogram of a Quadratic FM Chirp

The true position of the instantaneous frequency can be seen as a hyperbolic curve on the contour subplot of Figure 1.10(b). The quadratic FM Chirp is an even more dynamic signal than the linear FM chirp. The spectral lobe broadens, best seen in Figure 1.10 (b), as the signal's frequency change accelerates.

We want, therefore, to look at other spectral techniques which may moderate some of the inherent weaknesses of the Spectrogram.

II. INSTANTANEOUS POWER SPECTRUM

A. INTRODUCTION

The Instantaneous Power Spectrum (IPS) is based on a class of time-frequency distributions called "Cohen's class" [Ref. 3]. Cohen's class is defined for a continuous signal by

$$C_x(t,f) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \phi(\nu,\tau)x(\xi+\frac{\tau}{2})x^*(\xi-\frac{\tau}{2})e^{j2\pi(\nu\xi-\nu t-f\tau)}d\nu d\xi d\tau \qquad (2.1)$$

Many different power spectral techniques can be derived from this class of distributions including Wigner-Ville, IPS and Rihaczek time-frequency representations [Ref. 4]. Each of these techniques can be obtained from the generalized expression by the selection of an appropriate kernel, $\phi_c(\nu,\tau)$. In the case of IPS, the kernel used is $e^{j\pi\nu\tau}$. Originally the estimate of the instantaneous frequency content utilizing Page's definition of the instantaneous power spectrum as the derivative of a running spectrum [Ref. 8]:

$$\rho^-(t,f) = \frac{\partial}{\partial t}\left|S_t^-(f)\right|^2 \qquad (2.2)$$

where

$$S_t^-(f) = \int_{-\infty}^{t} s(\tau)e^{-j2\pi f\tau}d\tau. \qquad (2.3)$$

The concept of the instantaneous power spectrum was further expanded by Levin [Ref. 9] with the addition of a backward running spectrum defined as:

$$\rho^+(t,f) = \frac{-\partial}{\partial t}\left|S_t^+(f)\right| \qquad (2.4)$$

where

$$S_t^+(f) = \int_{t}^{\infty} s(\tau)e^{-j2\pi f\tau}d\tau. \qquad (2.5)$$

Using both the forward and backward running spectrums we can define IPS as the average of these two spectra given by

$$IPS(t,f)=\frac{1}{2}\left[\rho^{-}(t,f)+\rho^{+}(t,f)\right] \tag{2.6}$$

The discrete form of the IPS algorithm utilized in this thesis is [Ref. 5]

$$IPS_x(n,\omega)=\frac{1}{2}\sum_{k=1}^{N_x}\left\{x(n)x^{*}(n-k)+x^{*}(n)x(n+k)\right\}w(0)w(k)e^{-j\omega k} \tag{2.7}$$

where $w(k)$ is a window function, $x(n)$ is the data sequence and N_x is the length of the data sequence. The discrete IPS expression is actually the Discrete Fourier Transform (DFT) of

$$\left\{x(n)x^{*}(n-k)+x^{*}(n)(n+k)\right\}w(0)w(k) \tag{2.8}$$

In order to exploit the efficiency of the FFT the length of the data sequence is constrained to be a power of 2, such as 64, 256, or 512.

This chapter will explore three types of IPS programs which are given in the Appendix. The IPS program is designed for the analysis of relatively short data sequences, i.e., 64 to 1024 points. The IPSSURF program should be selected when data lengths are between 1024 and 2^{13} to 2^{15} points. The IPSLOFAR program should be selected to process data sets larger than 2^{15} points.

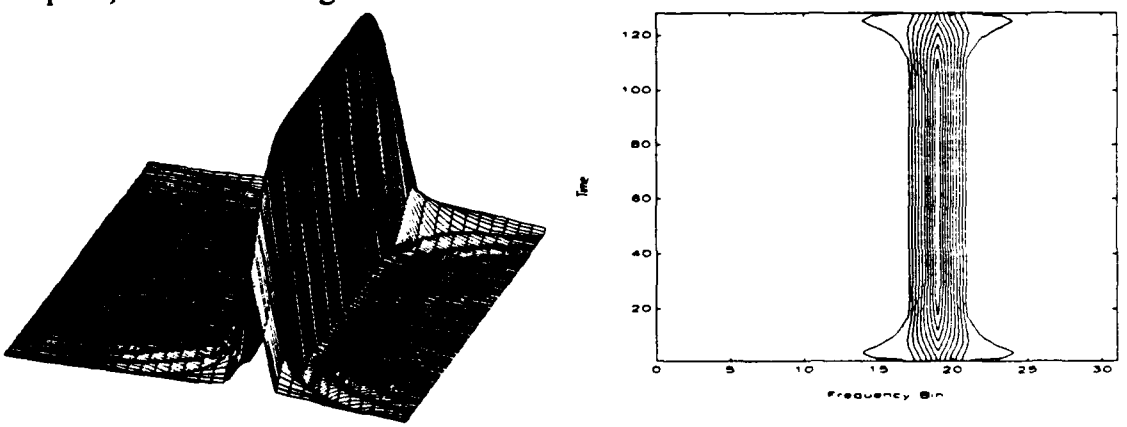
B. TRADITIONAL IPS TIME-FREQUENCY DISPLAY

For short data records, the traditional time-frequency display of the IPS algorithm is well-suited. Input parameters of the program define the dimensions of the returned time-frequency surface. The parameters include the window type (Rectangular or Hamming), the window length (normally half of the data sequence length) and the step

(the distance through which the window is stepped through the data sequence). As an example, if the data sequence is 512 points long with a window length of 256 points and a step of 1, the resulting surface contains 512 rows in the time direction having a frequency range of $-\pi$ to $+\pi$ divided into 256 frequency bins. The program assumes the data sequence to be of fixed duration. Since the IPS algorithm looks both backward and forward in time, the data sequence is padded with zeros at the beginning and end equal to the width of the selected window. The program returns only the positive frequency half of the resulting time-frequency surface to limit the size of the final display. In Figures 2.1 through 2.6, test data sets are used as signals of length 128. The window length is chosen to be 64 points with a Hamming window and the step is chosen to be 1.

1. Single Analytic Sinusoid

A single analytic sinusoid was created whose frequency location is exactly 19, i.e., the digital frequency was $\frac{19}{64}$. This implies that for a window length of 64, the spectral peak should occur at bin 19. Had the window length been 128 points, the frequency location would have been 38. The IPS surface, both mesh and contour subplots, are shown in Figure 2.1.



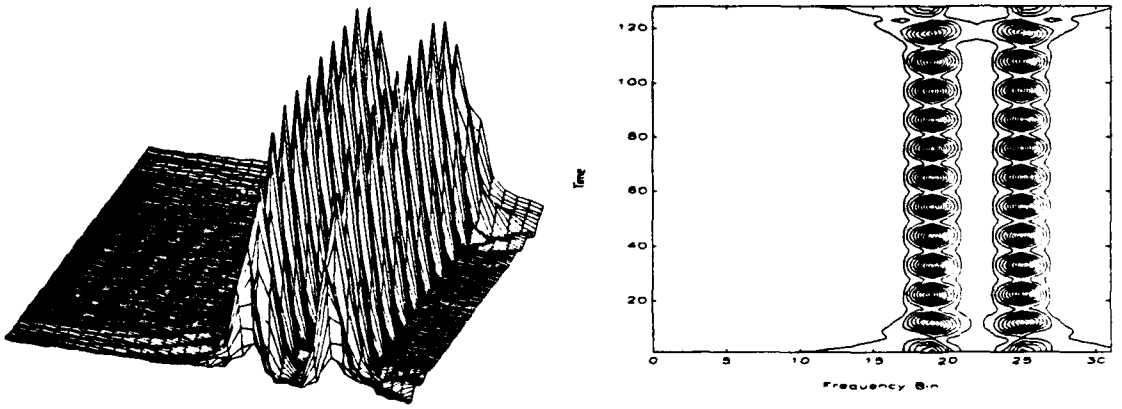
(a) (b)

Figure 2.1. Single Analytic Sinusoid via IPS

It can be seen that the frequency location is symmetrically centered at the proper location of 19. The width of the spectral ridge in Figure 2.1(a), measured at the point where the magnitude has dropped by 3 dB, is approximately three frequency bins wide. For this single analytic sinusoid data set, the sensitivity of the IPS algorithm is comparable to the Spectrogram. The IPS builds more quickly to a constant amplitude and trails off less quickly at the end of the data set than the Spectrogram would for the same data set. The sidelobes, evident on the mesh subplot, are never of such amplitude that they obscure the position of the spectral ridge and have dampened completely as the window moves well into the data set.

2. Multiple Analytic Sinusoids

A data set consisting of two analytic sinusoids was created with frequency locations of 19 and 25, respectively. The IPS surface, both mesh and contour subplots is shown in Figure 2.2.



(a) (b)

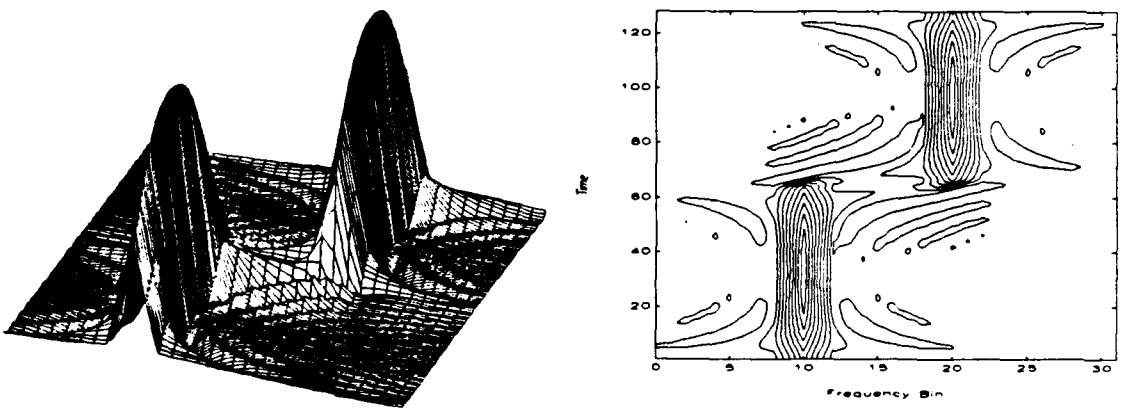
Figure 2.2. Multiple Analytic Sinusoid via IPS

It can be seen that the spectral ridges are symmetrically centered at the proper locations of 19 and 25 and their 3 dB widths are, as in Figure 2.1, still approximately three frequency

bins wide. The sidelobes, evident on the mesh surface, dampen as the window moves into the data set. The modulation, evident along the spectral ridges, is a consequence of cross-spectral terms which ride on the autocorrelation terms. One significant advantage of the IPS algorithm, compared to the Wigner-Ville algorithm, is that it does not experience spurious cross-spectral terms between the strong spectral peaks. The intra-ridge modulation of the IPS affects the display of the time-frequency surface but does not degrade or obscure the determination of the spectral locations.

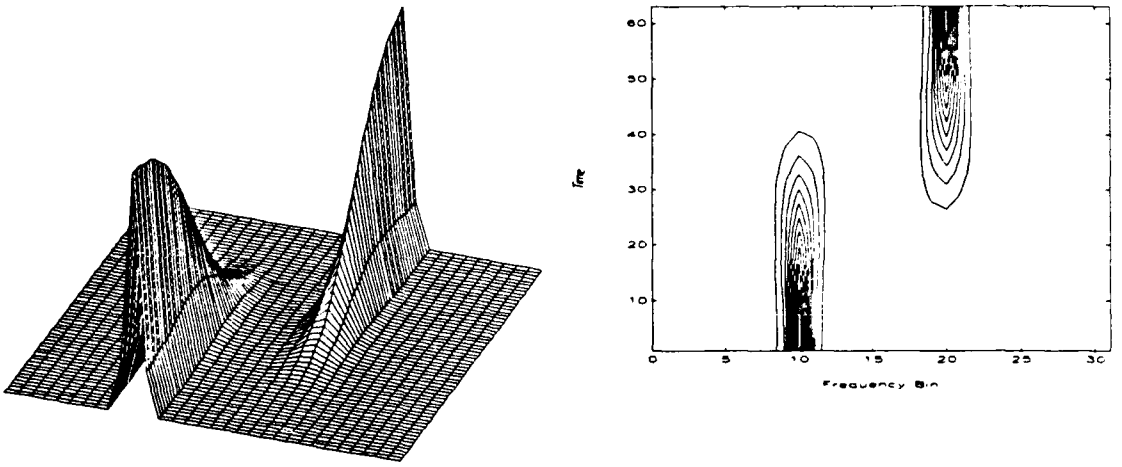
3. Frequency Shift Key (FSK)

A Frequency Shift Key (FSK) data set was created composed of an analytic sinusoid whose frequency was shifted from $\frac{10}{64}$ to $\frac{20}{64}$ midway through the data set of 128 points. The resultant IPS surface, both mesh and contour subplots is shown in Figure 2.3, the corresponding Spectrogram is shown in Figure 2.4.



(a) (b)

Figure 2.3. IPS of FSK data set



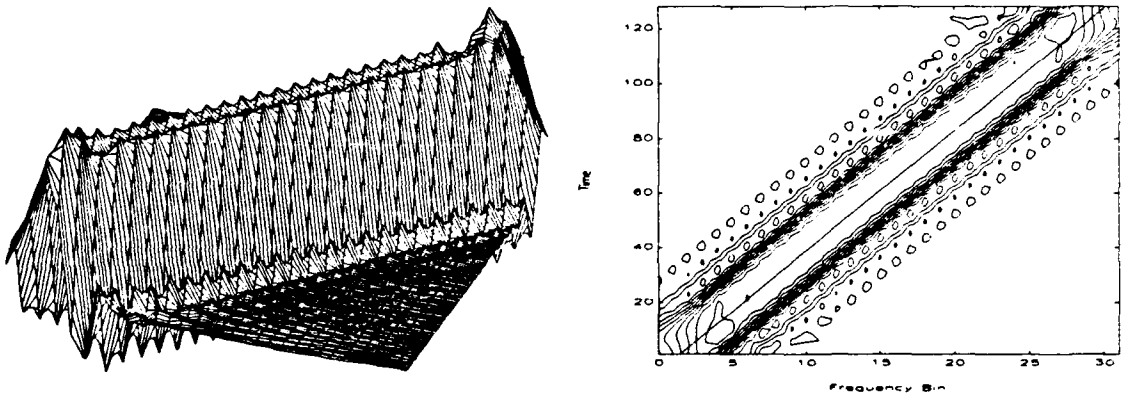
(a) (b)

Figure 2.4. Spectrogram of FSK data set

Both the IPS surface and the Spectrogram correctly locate the respective frequency locations, however the IPS surface accurately discerns the time of the signals. The IPS surface clearly shows that the frequency shifted at time 64 to a higher frequency. The Spectrogram incorrectly shows an overlap in the time of the two signals. The IPS surface does experience sidelobes which never completely disappear through its extent, however it can be argued that the correct spectral location in frequency is of more significance.

4. Linear FM Chirp

A data set consisting of a signal whose frequency changes linearly with time, i.e. linear FM chirp, was made to transition from a frequency of $\frac{1}{64}$ to $\frac{30}{64}$. The resultant IPS surface, both mesh and contour subplots, is shown in Figure 2.5.



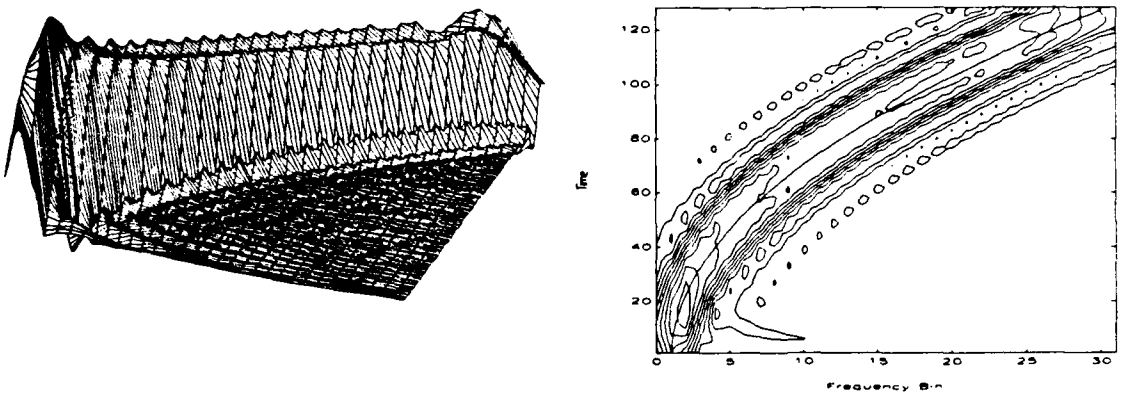
(a) (b)

Figure 2.5. IPS of Linear FM Chirp

The true position of the instantaneous frequency can be seen as a straight line on the contour subplot of Figure 2.5(b). The width of the spectral ridge is broader than the previous figures, however, drawing a line along the centerline of the structure would clearly correctly locate the instantaneous frequency at the correct time in the data set.

5. Quadratic FM Chirp

A data set consisting of a signal whose frequency changes as a quadratic function of time, i.e. over the length of the data the quadratic FM chirp, was made to transition from a frequency of $\frac{1}{64}$ to $\frac{30}{64}$. The resultant IPS surface, both mesh and contour subplots is shown in Figure 2.6.



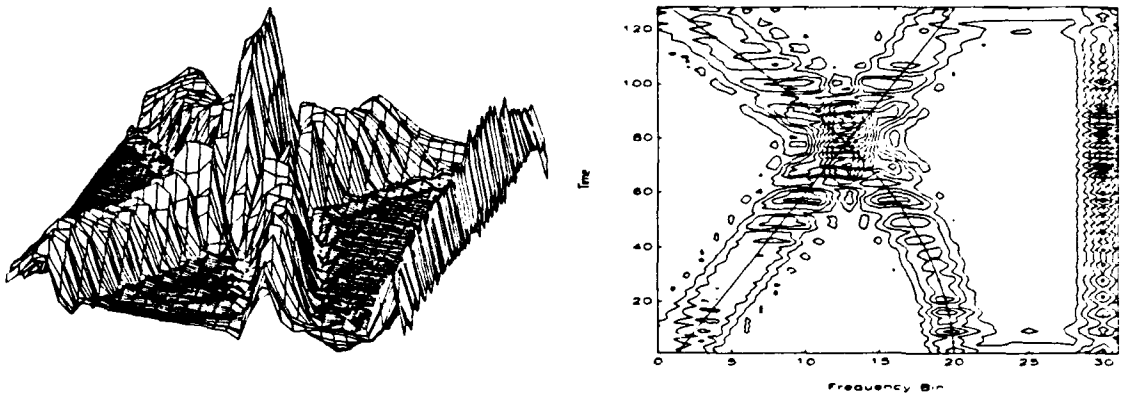
(a) (b)

Figure 2.6. IPS of a Quadratic FM Chirp

The true position of the instantaneous frequency is indicated by a hyperbolic line on the contour plot of Figure 2.6(b). The width of the spectral ridge is quite broad when compared to the stationary signal IPS surfaces, however, drawing a line along the centerline of the structure clearly locates the instantaneous frequency at the correct time in the data set. By comparing this IPS surface with Figure 1.10 Chapter I, the Spectrogram of this quadratic FM chirp data set, it can be seen that IPS maintains an almost constant spectral ridge width through the data set while the Spectrogram's spectral ridge broadens significantly and the apparent signal's frequency content changes rapidly.

6. Analytic Linearly Increasing FM Chirp, Quadratically Decreasing FM Chirp and stationary Sinusoid (Multiple Component Data Set)

A data set composed of three different signals was created. One of the signals is a linearly increasing FM chirp made to transition from frequency location 2 to 20; a quadratically decreasing FM chirp made to transition from frequency location 20 to 2 and an analytic sinusoid at frequency location 30. The resultant IPS surface, both mesh and contour subplots is shown in Figure 2.7.



(a) (b)

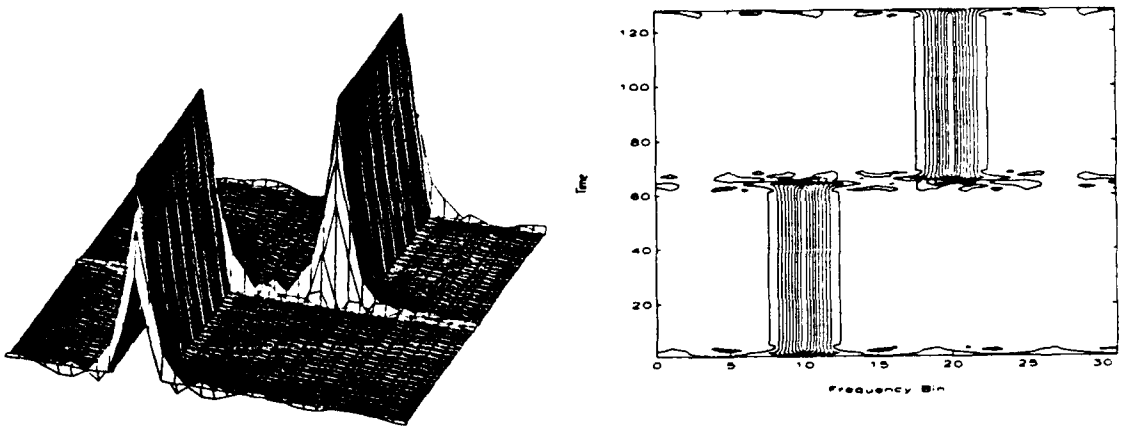
Figure 2.7. IPS of Multiple Component Signal

The correct locations of the instantaneous frequency for both the linear FM chirp and quadratic FM chirps are indicated by lines traced through the center of those structures, respectively. The presence and correct frequency locations of all three component signals of the data set is clearly shown. The area of intersection between the linear and quadratic chirps is broadened by the summing of the magnitudes of each signal. The width of each spectral lobe limits the resolving power of the IPS algorithm, however after comparison with Figures 2.1, 2.5 and 2.6, it can be seen that the width of the respective spectral lobes has not been affected by the presence of the other signal components. Also, cross terms between the true spectral locations are not observable.

C. LINKED IPS SURFACES

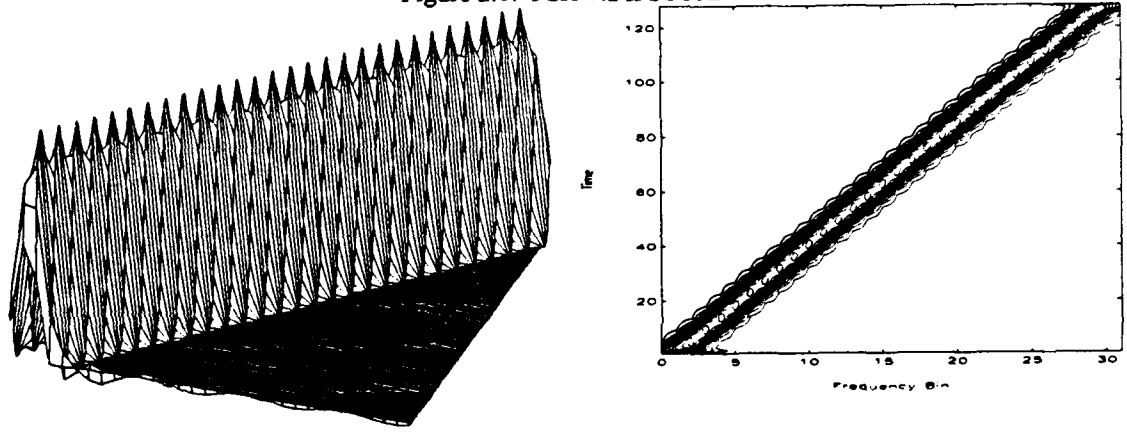
For larger data sets, i.e., more than 1024 points, the linked IPS time-frequency surface is a reasonable alternative to the IPS surface. As an example, if the data set were 2048 points long and the IPS program were invoked with a window length of 1024 points and a step of 1, the surface would be 2048 x 512. Such a surface would have 1,048,576 elements and with 8 bytes required to store an element, would require 8.4 Mbytes of computer storage! The IPSSURF program, in the Appendix, allows a larger data set to be more conveniently examined by dividing it into smaller segments, calculating a surface for

each of the smaller segments and then concatenating the surfaces together into a larger surface. Recall that the IPS program pads the data sequence with zeros to allow the algorithm to move backward and forward in time. The IPSSURF program does pad the first and last smaller data segments with zeros but the other data segments are padded with true past and future data points from the full data sequence. The IPSSURF parameters include the window type (Rectangular or Hamming), siglen (the desired length of the smaller data segments), the window length (normally half of the smaller data segment length) and the step (the distance through which the window is moved through the smaller data segments). As an example, if the original data sequence is 2048 points long and smaller data segments of 512 were selected with a window length of 256 points and a step of 8, the resulting surface contains 256 rows in the time direction having a frequency range of $-\pi$ to $+\pi$ divided into 256 frequency bins. The IPSSURF program, if invoked with a step size of 1, would concatenate full IPS surfaces into a very large surface. For this reason, the IPSSURF program is designed to be used as a broader analysis tool for an overall look of a large data sequence by invoking it with a step size larger than 1. Once an area of interest has been isolated with the IPSSURF program, finer analysis would be done using the IPS program. The test signals created for Figures 2.8 to 2.11, were 1,024 points long, the smaller data segment length (the siglen parameter) was selected as 128 points, the window length was selected as 64 points and the step was selected as 8. The IPSSURF subplots for FSK signal (Figure 2.8); Linear FM chirp (Figure 2.9); Quadratic FM chirp (Figure 2.10) and the Multiple Component Signal (Figure 2.11) follow. Each of the data sets were created as defined in sections B.3 through B.6.



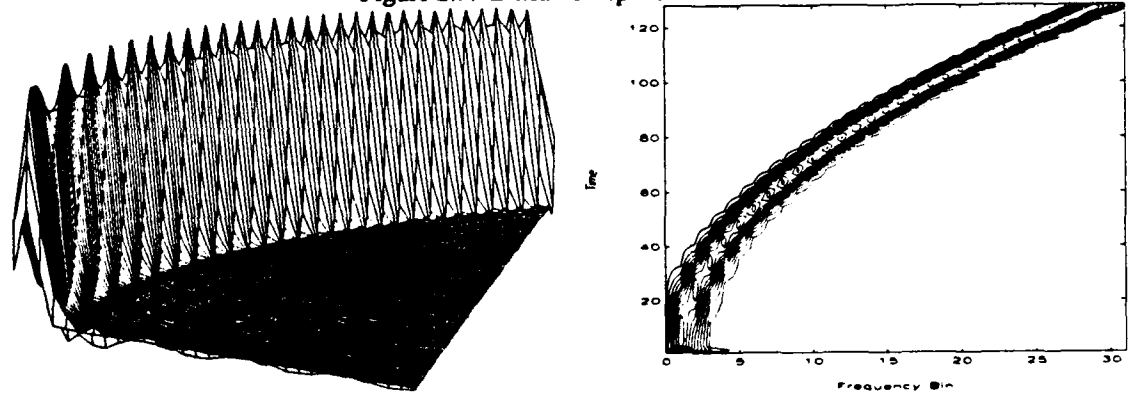
(a) (b)

Figure 2.8. FSK via IPSSURF



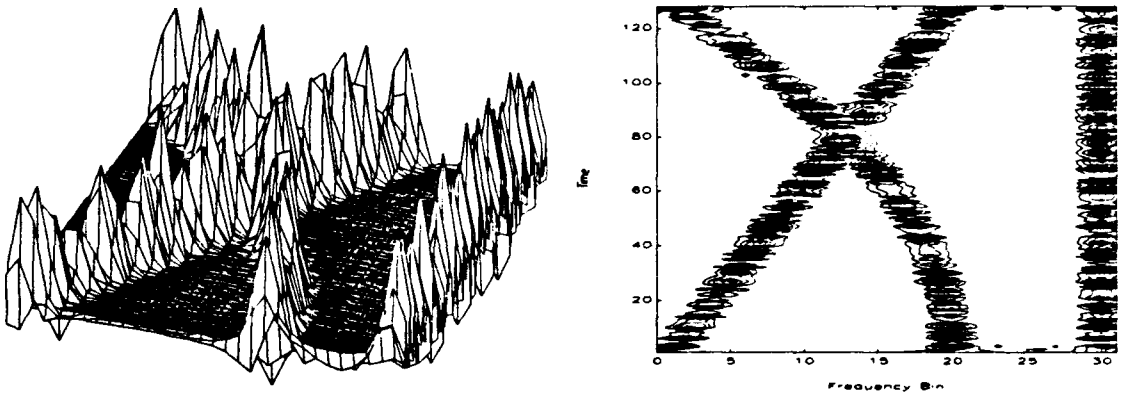
(a) (b)

Figure 2.9. Linear Chirp via IPSSURF



(a) (b)

Figure 2.10. Quadratic FM Chirp via IPSSURF



(a) (b)

Figure 2.11. Multi-component Signal with IPSSURF

Comparison of Figures 2.8 through 2.11 with the corresponding IPS Figures 2.3, 2.5 through 2.7 seem to indicate that the IPSSURF program shows finer spectral ridge details than the IPS surfaces. It can also be seen that the sidelobes on the IPSSURF surfaces seem to dampen more quickly than the IPS surfaces. These effects, however, are a result of time compression, determined by the selection of the step size parameter. As the step size is increased, finer details of the spectral surface would be progressively obscured. For this reason, the IPSSURF program should be used as a coarse analysis tool to locate and define areas of a large data set which could be further analyzed with the IPS program.

D. IPSLOFAR

The IPSLOFAR program (Appendix) is for very large data sets. IPSLOFAR is based on a "waterfall" display routinely used in the display of sonar data, called the LOFARGRAM. The IPSLOFAR program, like the IPSSURF program, divides a large data set into smaller segments and calculates IPS surfaces for each of the smaller segments. Unlike the IPSSURF program, the average over time is then calculated and

placed as a row in a larger surface for display. The IPSLOFAR program can be seen as a part of the continuum of 'IPS' programs where the IPS program is the finest analysis tool and the IPSLOFAR program is the coarsest analysis tool. The IPSLOFAR program parameters include the window type (Rectangular or Hamming), siglen (the desired length of the smaller data segments), the window length (normally half of the smaller data segment length) and the step (the distance through which the window is stepped through the smaller data segment).

A data set was created of length 16,384 points. The signal is composed of a linearly decreasing FM chirp made to transition from frequency location 30 to 1, a two component FSK signal which switched from frequency location 10 to 20 midway through the data sequence and an analytic sinusoid at frequency location 30. The IPSLOFAR parameters used were a siglen of 128 points, a window length of 64 points, a Hamming window and a step of 8. The IPSLOFAR contour plot is shown in Figure 2.12(a) and a plot of the average over time of the surface is shown in Figure 2.12(b).

The IPSLOFAR display clearly shows all signal components. The width of the spectral ridges is comparable to both the IPS and IPSSURF programs. The Lofargram, used in sonar displays, is actually an intensity plot, vice a contour plot. Because we can only use contour plots for IPSLOFAR in Matlab, the display is somewhat different from a traditional Lofargram.

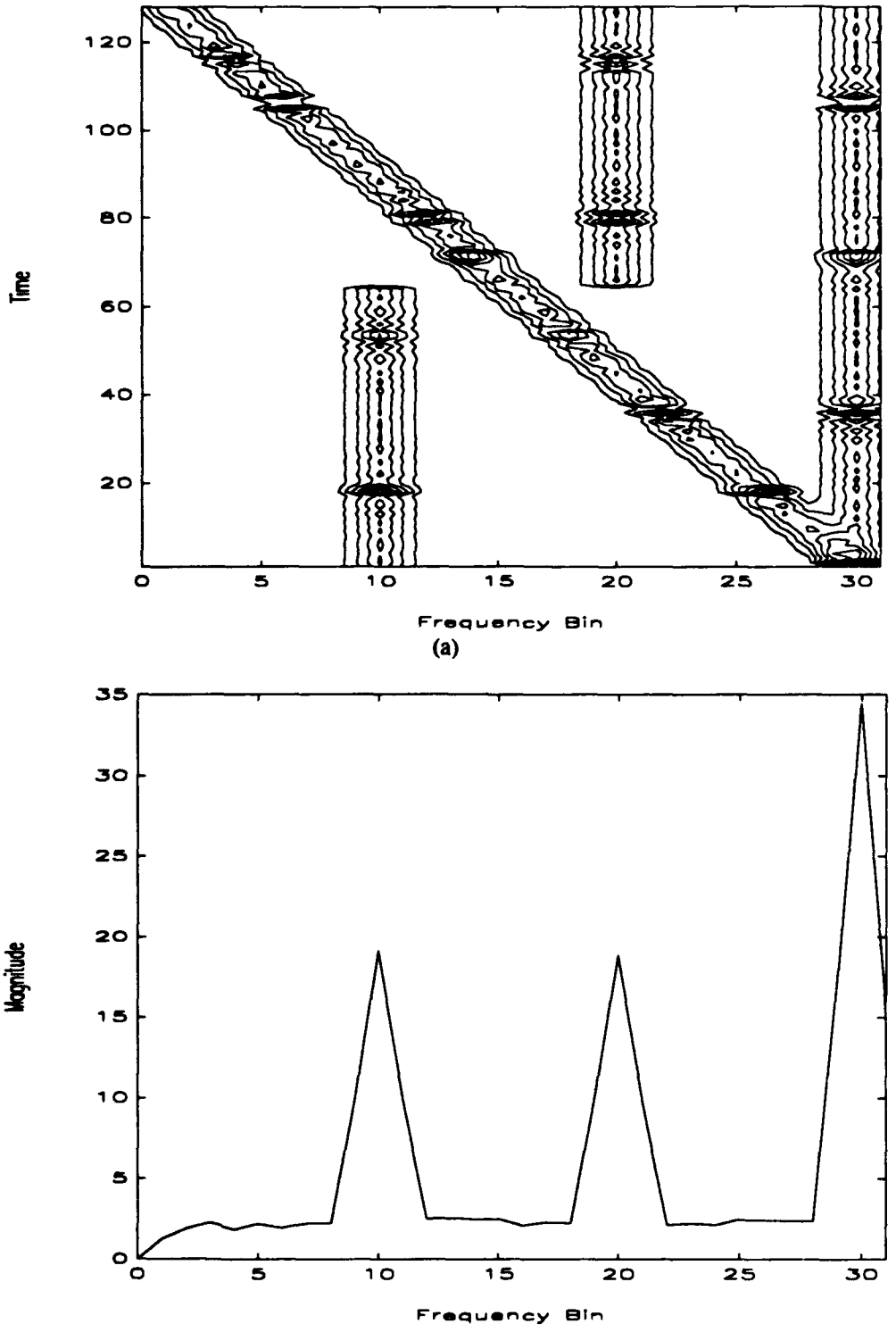


Figure 2.12. IPSLOFAR surface for a Multi-component Signal

III. 1 ½ D INSTANTANEOUS POWER SPECTRUM

A. INTRODUCTION

Both the Spectrogram and IPS techniques are second moment spectral analysis techniques. They and other related techniques are well-suited to extract spectral information, however when additive Gaussian noise perturbs signals, the application of higher order moments is thought to improve the signal to noise performance of the spectral algorithm. For zero mean Gaussian random processes, the third moment and third order cumulant are equal to zero. The 1 ½ D instantaneous power spectrum [Ref. 6] is based on a third order cumulant. Cumulants are related to ordinary moments, as a matter of fact, for a zero mean random process the first and second order cumulants are equal to the first and second moments. For higher order spectral techniques, the use of cumulants is often preferred to higher order moments because cumulants can measure the departure of a random process from a Gaussian random process [Ref. 3]. The 1 ½ D instantaneous power spectrum implemented in this thesis is derived from the bispectrum. The bispectrum is a third order spectrum defined as [Ref. 3]

$$S_x^{(3)}(e^{j\omega_1}, e^{j\omega_2}) = \sum_{l_1=-\infty}^{\infty} \sum_{l_2=-\infty}^{\infty} C_x^{(3)}[l_1, l_2] e^{-j(\omega_1 l_1 + \omega_2 l_2)}; \tag{3.1}$$

where $C_x^{(3)}[l_1, l_2]$ is the third order cumulant. If we assume that we have a zero-mean random process the third order cumulant is

$$C_x^{(3)}[l_1, l_2] = \mathrm{E}\{x^*(n)x(n+l_1)x(n+l_2)\}. \tag{3.2}$$

We can derive a degenerate estimate of this cumulant by setting l_1 to zero, therefore

$$C_x^{(3)}[0, l_2] = \mathrm{E}\{x^*(n)x(n)x(n+l_2)\} \tag{3.3}$$

and

$$C_x^{(3)}[0,l_2] = \mathrm{E}\left\{|x(n)|^2 x(n+l_2)\right\}. \qquad (3.4)$$

Furthermore, when we replace the expectation with the instantaneous value the third order spectrum becomes

$$S_x(e^{j\omega}) = \sum_{k=-\infty}^{\infty} |x(n)|^2 x(n+k) e^{-j2\pi\omega k} \qquad (3.5)$$

As in the derivation of the IPS technique of Chapter II, we will utilize both Page's definition of the instantaneous power spectrum and Levin's addition of a backward running spectrum to finally derive the discrete form of the 1 ½ D instantaneous power spectrum used in this thesis as

$$1\ \tfrac{1}{2}\ \mathrm{D}(n,\omega) = \frac{1}{2}\sum_{k=1}^{N_x}\left\{|x(n)|^2 x(n-k) + |x(n)|^2 x(n+k)\right\} w(0)w(k)e^{-j\omega k}; \qquad (3.6)$$

where $w(k)$ is a window function, $x(n)$ is the data sequence and N_x is the length of the data sequence. As in the IPS technique, the length of the data sequence is constrained to be a power of 2, such as 64, 256, or 512.

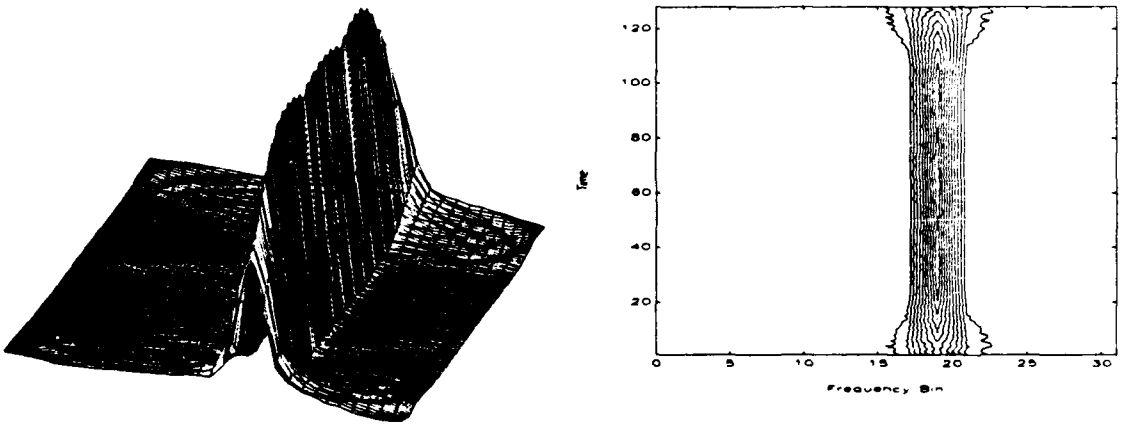
The three 1 ½ D programs are given in the Appendix. The ONE_HALF program is used for relatively short data sequences, i.e., 64 to 1024 points. The ONESURF program is used for data sequences typically between 1024 and 2^{13} and 2^{15} points. The ONELOFAR program should be used for data sets larger than 2^{15} points. The philosophy of the 1 ½ D programs is analogous to the corresponding IPS programs of Chapter II.

B. TRADITIONAL 1 ½ D TIME-FREQUENCY DISPLAY

The parameters for the ONE_HALF program include the window type (Rectangular or Hamming), the window length (normally half of the data sequence length) and the step

(the distance through which the window is stepped through the data sequence). The program returns only the positive frequency half of the resulting time-frequency surface to limit the size of the final display. For Figures 3.1 through 3.6, test data sets of length 128 were created. The window length is chosen to be 64 points with a Hamming window and the step is chosen to be 1. Figures 3.1 through 3.6 utilize the same data sets as were used in Figures 2.1 through 2.3 and 2.5 through 2.7 of Chapter II, respectively.

1. Single Analytic Sinusoid

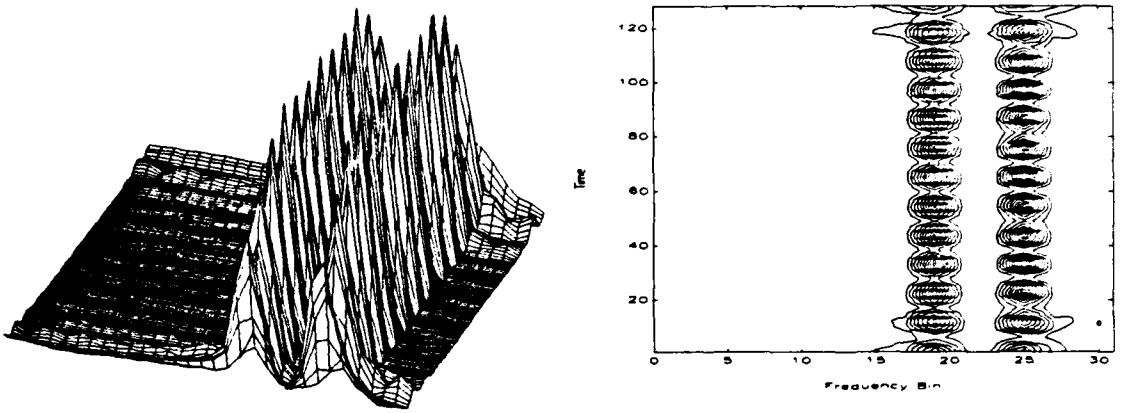


(a) (b)

Figure 3.1. Single Analytic Sinusoid via 1 ½ D

When compared to Figure 2.1, the appearance of the spectral ridge on the mesh subplot of Figure 3.1, is not as smooth in appearance. The width of the spectral ridge measured at the point where the magnitude has dropped by 3 dB is comparable to the IPS technique. The sidelobes on the time-frequency surface are less defined but the extent of their effect is again comparable to the IPS technique. It can also be seen that the 1 ½ D technique quickly reaches a constant amplitude and falls off in times comparable to the corresponding IPS case. Also during the central region of the surface, along the time axis, the noise background seems to be smaller in Figure 3.1 than the corresponding Figure 2.1.

2. Multiple Analytic Sinusoids

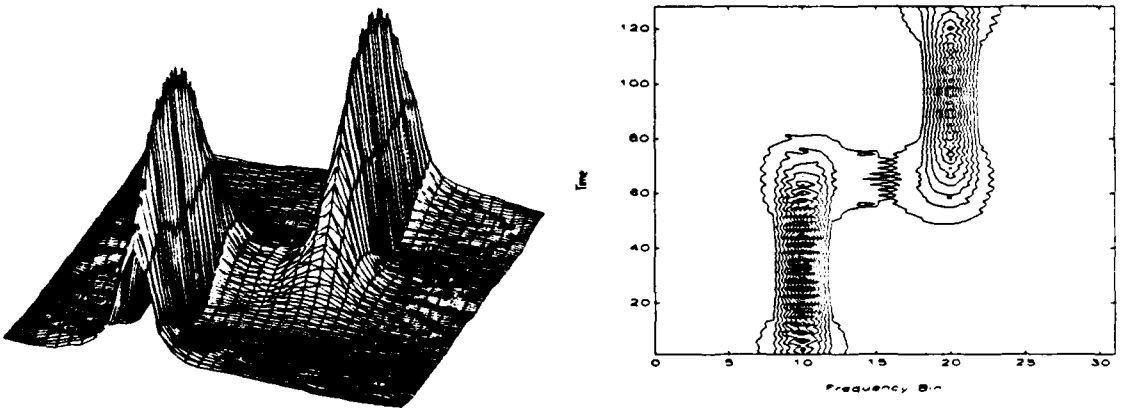


(a) (b)

Figure 3.2. Multiple Analytic Sinusoid via 1 ½D

When compared to the corresponding IPS surface, Figure 2.2, the 1 ½ D time-frequency peaks at the true frequency locations having comparable spectral ridge widths. The envelope of the spectral ridges fluctuates more than on the corresponding IPS surface.

3. Frequency Shift Key (FSK)

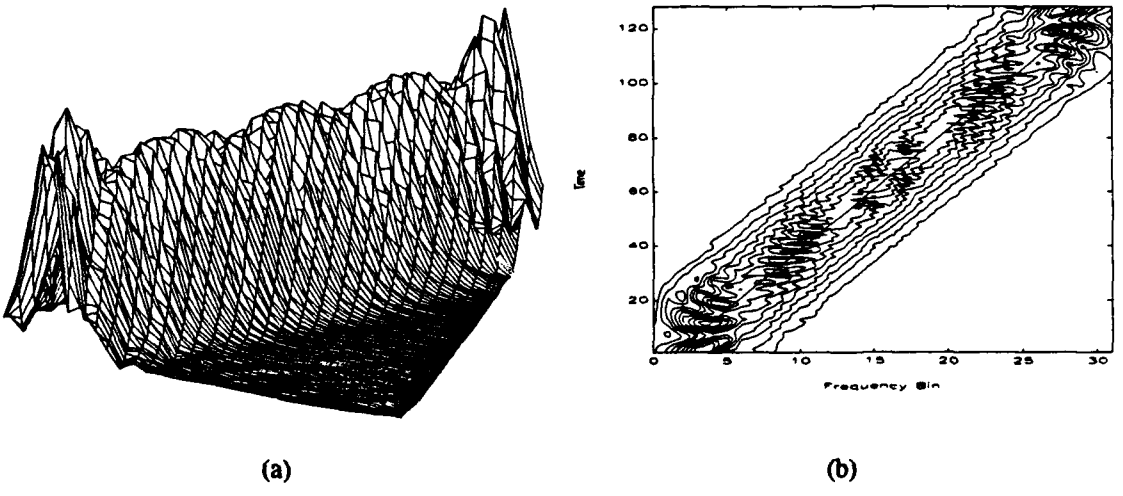


(a) (b)

Figure 3.3. FSK signal via ONE_HALF

For the FSK signal data set, areas in the 1 ½ D surface where the frequency shift occurs differs from the corresponding IPS surface of Figure 2.3. The 1 ½ D technique does not delineate the time of the signals as accurately as the IPS technique. The time of the frequency shift is obscured by the appearance of cross terms.

4. Linear FM Chirp

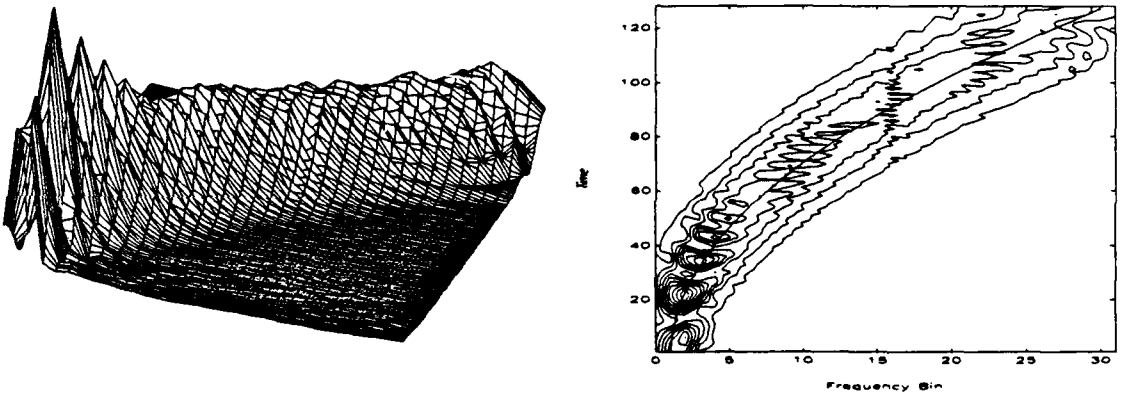


(a) (b)

Figure 3.4. IPS of Linear FM Chirp

The true position of the instantaneous frequency can be seen as a straight line on the contour subplot of Figure 3.4(b). The width of the 1 ½ D spectral ridge is wider than the corresponding IPS surface in Figure 2.5. The surface has a coarser appearance, showing less detail, than the corresponding IPS surface. The envelope of the spectral ridge shows more variability with slower intra-ridge modulation (i.e., fluctuations along the ridge top are at a lower frequency) than the corresponding IPS surface.

5. Quadratic FM Chirp

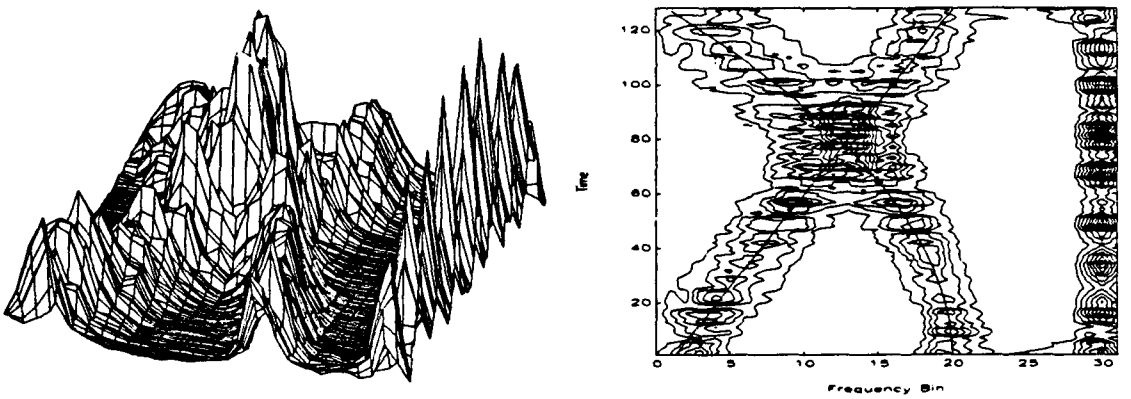


(a) (b)

Figure 3.5. IPS of a Quadratic FM Chirp

The true position of the instantaneous frequency is indicated by a hyperbolic line on the subplot of Figure 3.5(b). Once again, when compared with the IPS surface in Figure 2.6, it can be seen that the width of the spectral ridge is broader than in the IPS case, however the true instantaneous frequency can be obtained as an average along the spectral ridge. It can also be seen that the amplitude of the spectral ridge decreases towards the end of the data set and also broadens. This example shows that the 1 ½ D is less suited for a dynamic signal than IPS but better suited than the spectrogram, if one disregards issues such as noise.

6. Analytic Linearly Increasing FM Chirp, Quadratically Decreasing FM Chirp and stationary Sinusoid (Multiple Component Data Set)



(a) (b)

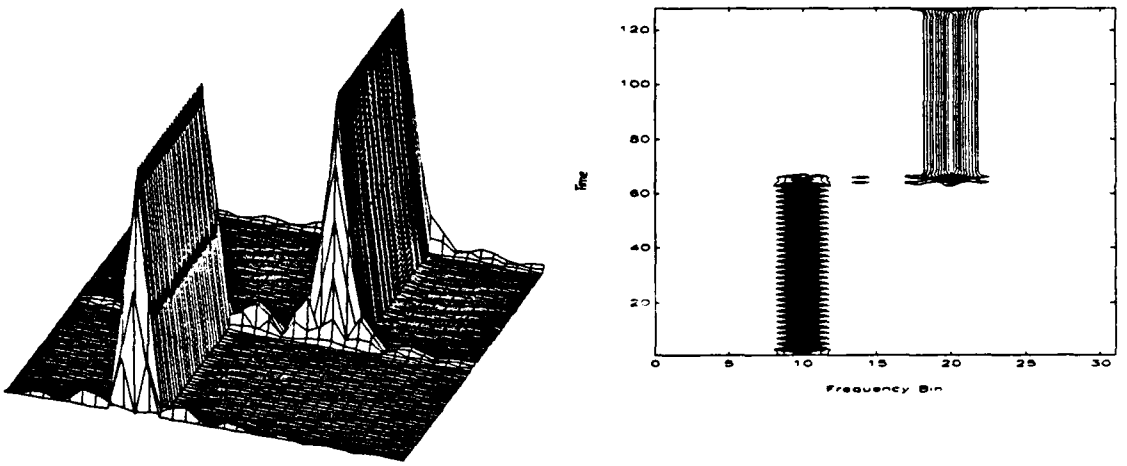
Figure 3.6. IPS of Multiple Component Signal

When compared with the IPS surface in Figure 2.7, the width of the spectral ridges indicating the dynamic signal components, i.e., the linear FM chirp and the quadratic FM chirp is wider than IPS. The width of the spectral ridge which indicates the stationary sinusoid is comparable to the IPS surface, although there is more modulation apparent on the spectral ridge.

C. LINKED 1 ½ D SURFACES

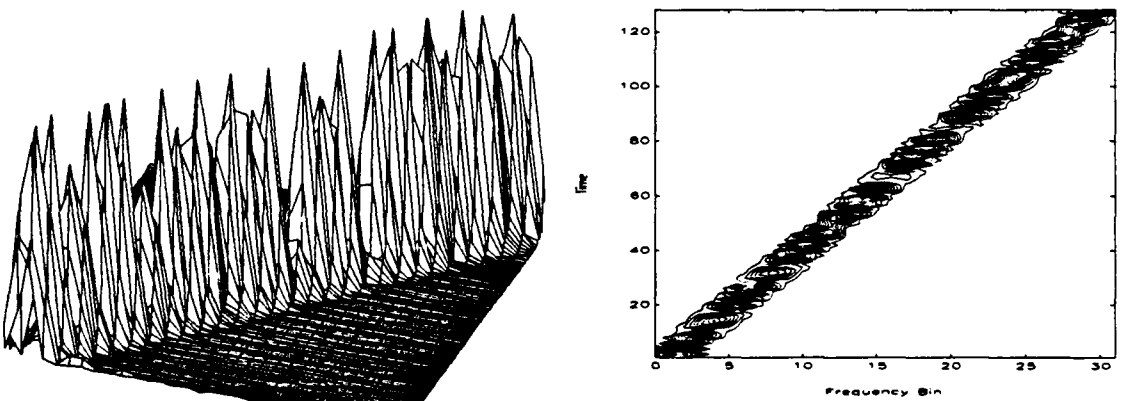
The parameters for the ONESURF program include the window type (Rectangular or Hamming), siglen (the desired length of the smaller data segments), the window length (normally half of the smaller data segment length) and the step (the distance through which the window is moved through the smaller data segments). The ONESURF program is designed to be used as a broader analysis tool for an overall look of a large data sequence by invoking it with a step size larger than 1. The ONESURF program is used to plot the 1 ½ D surface of an FSK signal (Figure 3.7); a linear FM chirp (Figure 3.8); a

quadratic FM chirp (Figure 3.9) and a multiple component signal. Each of the data sets were created as defined in Chapter II, sections B.3 through B.6.



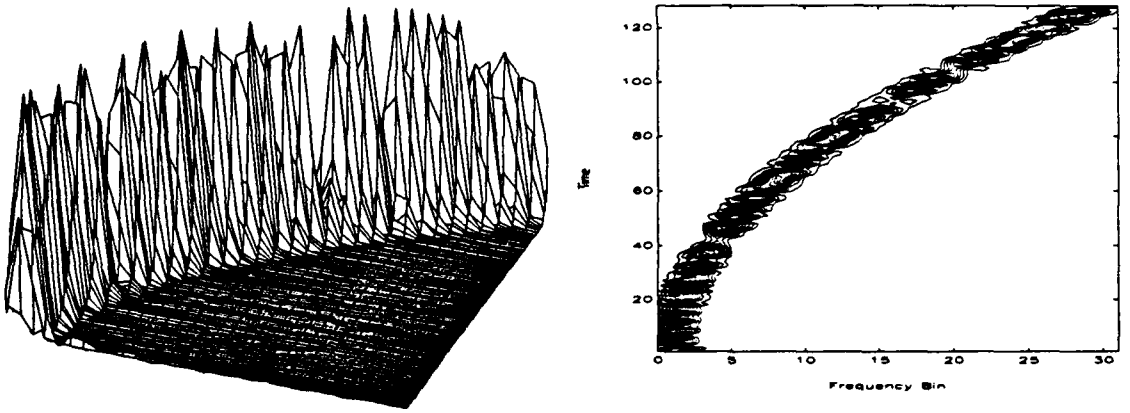
(a) (b)

Figure 3.7. FSK via ONESURF



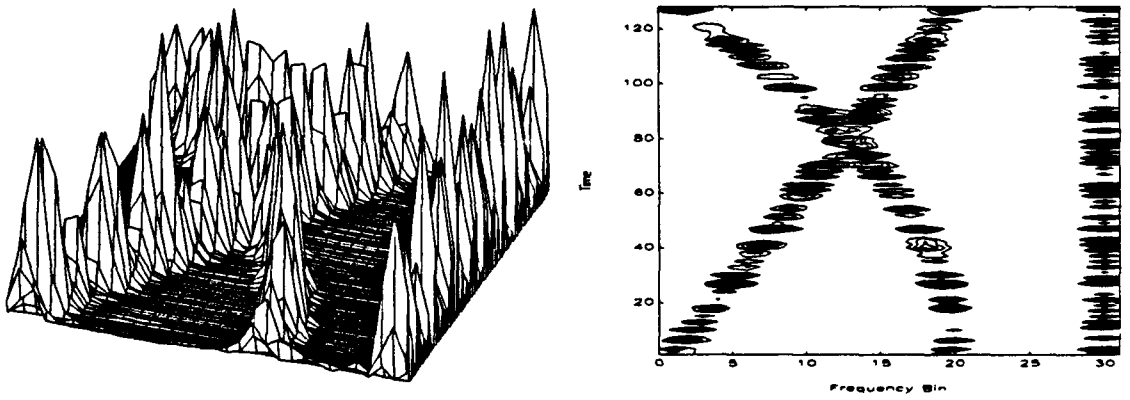
(a) (b)

Figure 3.8. Linear Chirp via ONESURF



(a) (b)

Figure 3.9. Quadratic FM Chirp via IPSSURF



(a) (b)

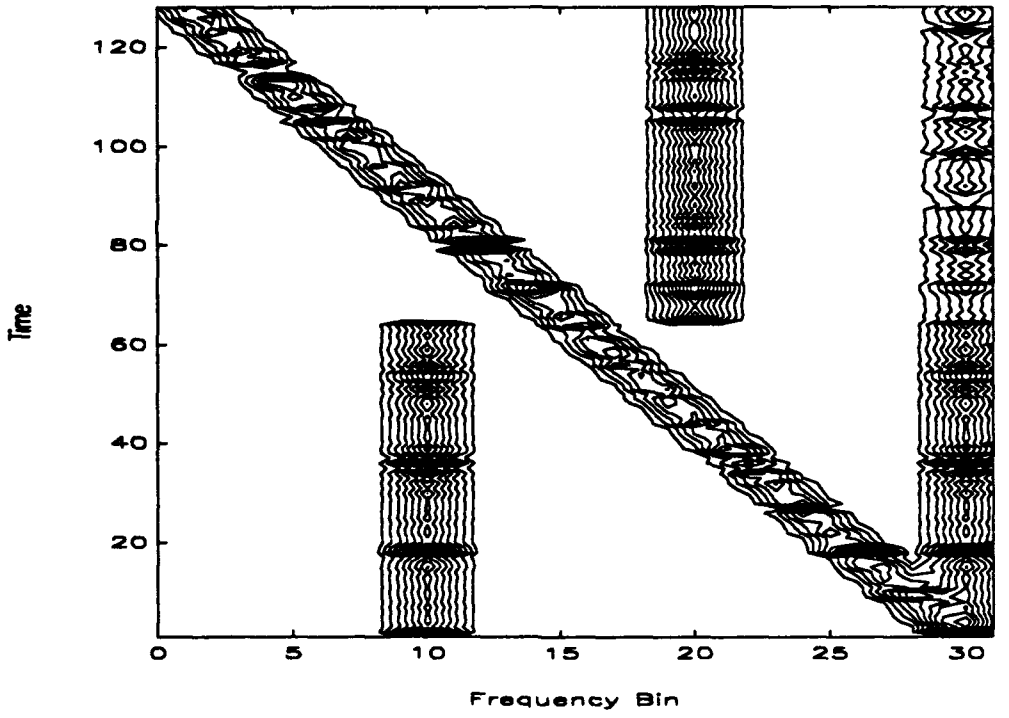
Figure 3.10. Multi-component Signal with IPSSURF

Comparison of Figures 3.7 through 3.10 with the corresponding IPSSURF surfaces in Chapter II, Figures 2.8 through 2.11 shows very few differences. The ONESURF program, as the IPSSURF program, is best suited as a coarse analysis tool to locate and define areas of a large data set which could be further analyzed with the ONE-HALF program.

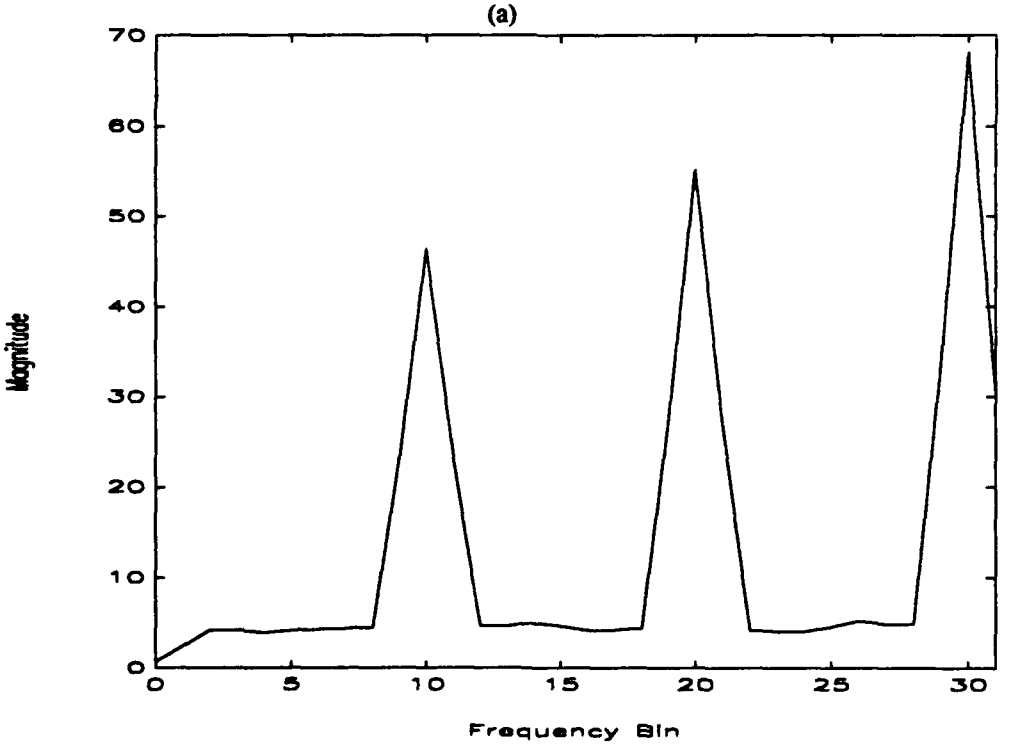
D. ONELOFAR

The ONELOFAR program parameters include the window type (Rectangular or Hamming), siglen (the desired length of the smaller data segments), the window length (normally half of the smaller data segment length) and the step (the distance through which the window is stepped through the smaller data segment).

The experimental test signal used is the same as used in the IPS chapter, section D. The ONELOFAR parameters used are a siglen of 128 points, a window length of 64 points, a hamming window and a step of 8. The ONELOFAR contour plot and corresponding plot of the average over time of the surface is shown in Figure 3.11(a) and (b), respectively. The ONELOFAR display clearly shows all signal components. The plot of the average over time also helps to locate the spectral components of the signal, especially in low SNR environments. That is, it can be used to extract information at very low SNR's, provided the spectral components are stationary.



(a)



(b)

Figure 3.11. ONELOFAR surface for a Multi-component Signal

IV. PERFORMANCE AND COMPARISON OF SPECTROGRAM, INSTANTANEOUS POWER SPECTRUM AND 1 ½ D TECHNIQUES

A. INTRODUCTION

Chapters I, II and III discussed, respectively, the Spectrogram, IPS and 1 ½ D techniques. Several test data sets were created and subsequently analyzed. Now we investigate the strengths and the weaknesses of each of these techniques in relation to one another with signals embedded in Gaussian white noise. A typical ocean acoustic data set is also processed.

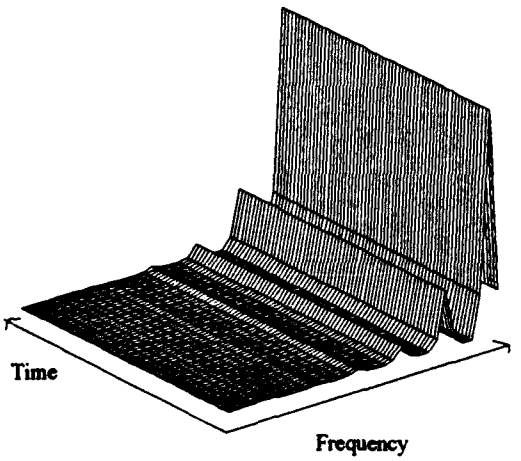
B. SPECTRAL SENSITIVITY WITH SINUSOIDS IN ADDITIVE GAUSSIAN WHITE NOISE

A data set was created, composed of sinusoids at frequency locations 5, 10, 15, 20, 25 and 30 which were embedded, respectively, in additive Gaussian white noise at -6, -3, 0, 3, 6 and 9 dB SNR. Figures 4.1 through 4.3 show the respective Spectrogram, IPS and ONE_HALF (1 ½ D) time-frequency surfaces in subplots using

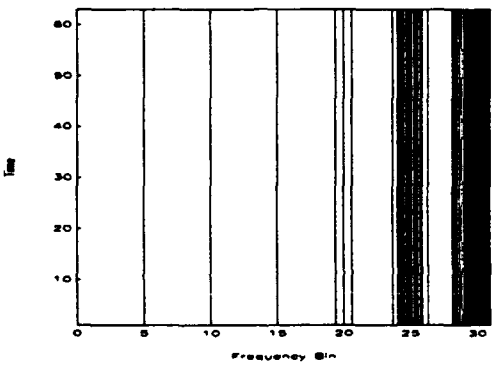
- (a) mesh surface
- (b) contour plot with the true frequency locations shown by a line at each location
- (c) plot of the average over time of the corresponding time-frequency surface.

The integration time for the spectrogram was 64 while the window length for the IPS and 1 ½ D programs was also 64. The mesh subplots for both the IPS and 1 ½ D techniques show their characteristic modulation along the spectral ridges due to the cross-spectral terms. The Spectrogram mesh surface has no modulation along its spectral ridge, which is also characteristic for this technique. The respective contour plots of the three techniques offer more information. It can be seen that the width of the spectral ridge

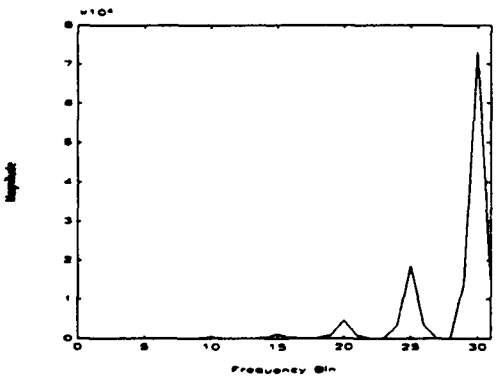
is narrower in the Spectrogram compared to the other two techniques. However, it can also be seen that where the Spectrogram can locate the sinusoid in the Gaussian white noise to down to 3 dB SNR, the IPS techniques can locate down to -3 dB SNR and the 1 ½ D technique even locates to -6 dB SNR.



(a)

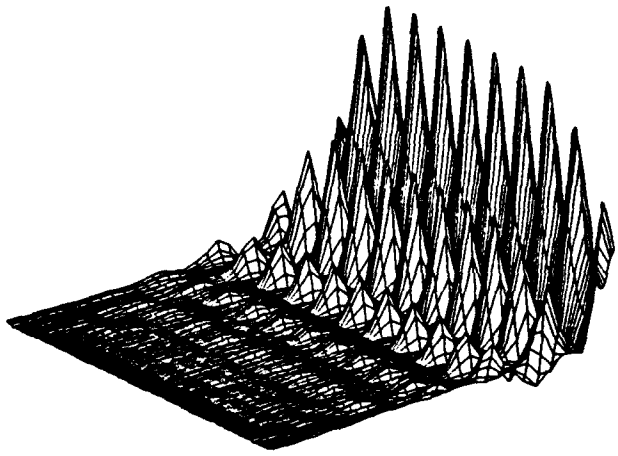


(b)

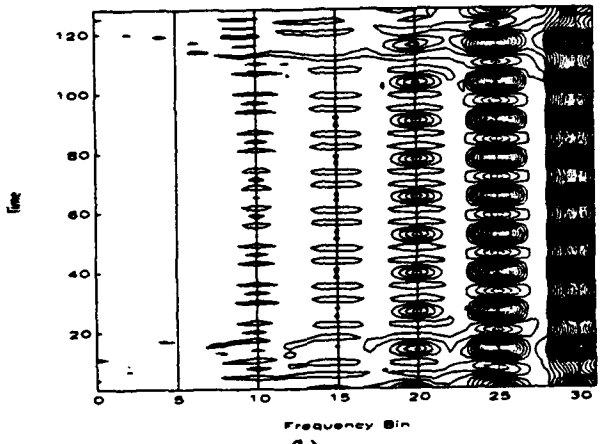


(c)

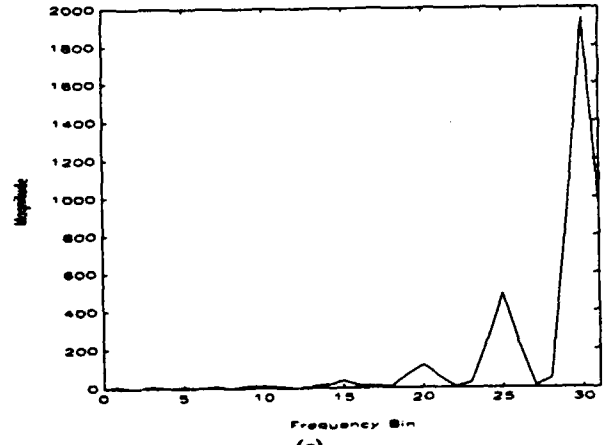
Figure 4.1. Spectrogram of Sinusoids embedded in Gaussian White Noise at -6, -3, 0, 3, 6 and 9 dB SNR respectively at frequency locations 5, 10, 15, 20, 25 and 30



(a)

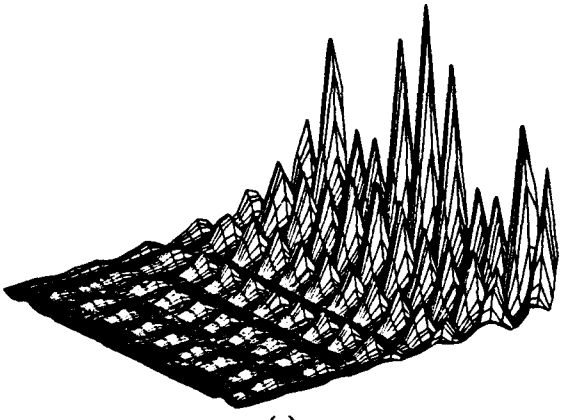


(b)

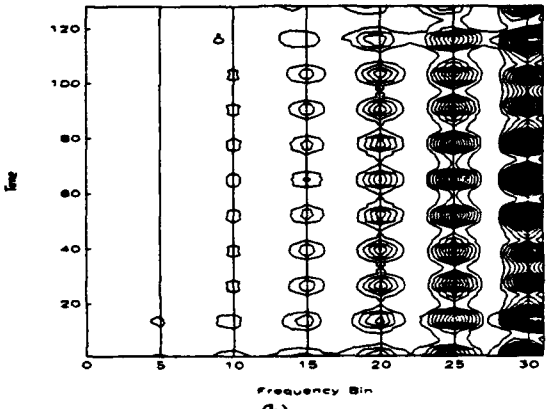


(c)

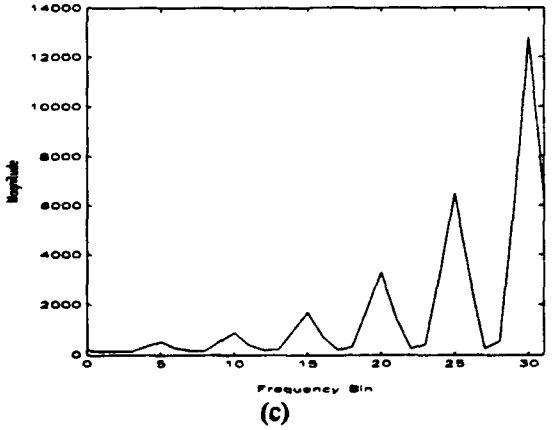
Figure 4.2. IPS of Sinusoids embedded in Gaussian White Noise at -6, -3, 0, 3, 6 and 9 dB SNR respectively at frequency locations 5, 10, 15, 20, 25 and 30



(a)



(b)



(c)

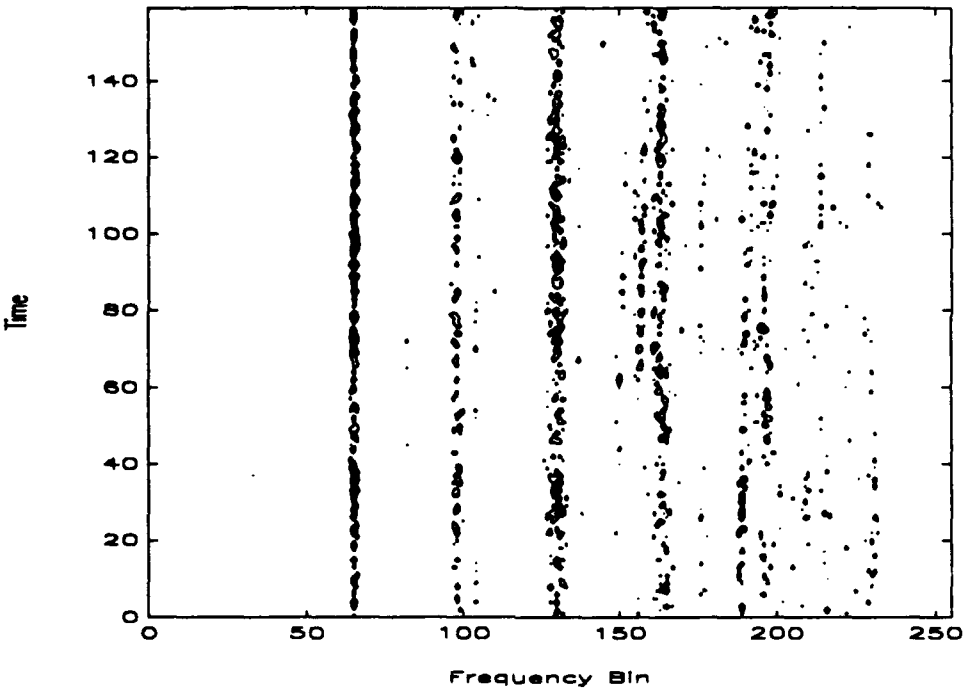
Figure 4.3. 1 ½ D of Sinusoids embedded in Gaussian White Noise at -6, -3, 0, 3, 6 and 9 dB SNR respectively at frequency locations 5, 10, 15, 20, 25 and 30

C. SPECTRAL SENSITIVITY WITH AN OCEAN ACOUSTIC DATA SET

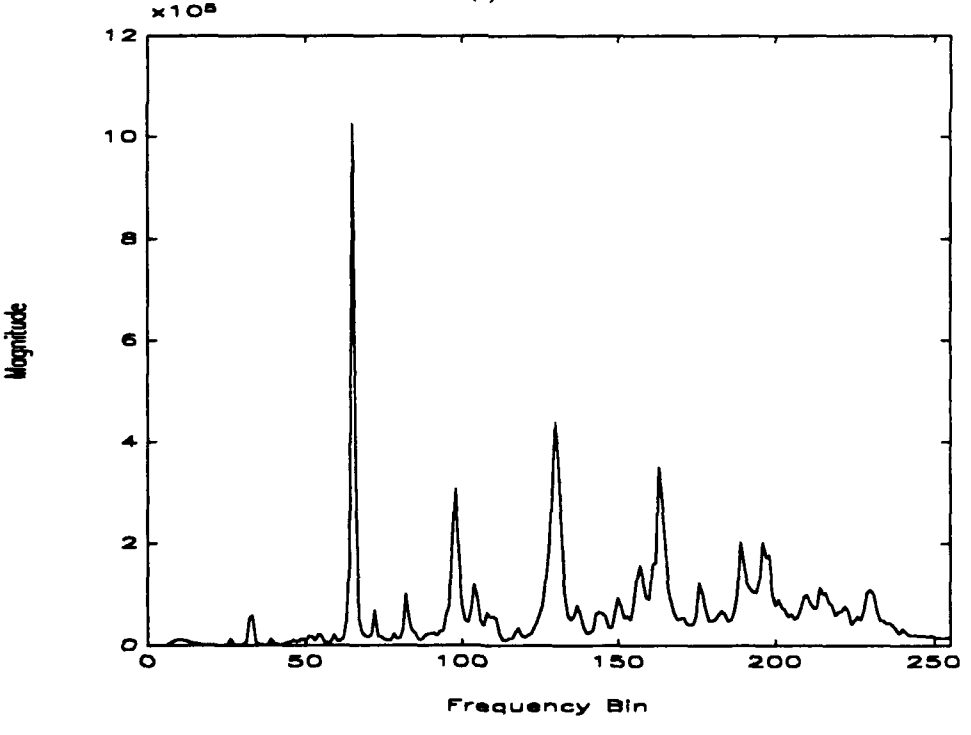
Figures 4.4 through 4.6 show the respective Spectrogram, IPS and ONE_HALF (1 ½ D) time-frequency surfaces in subplots using

- (a) contour plot with the true frequency locations shown by a line at each location
- (b) plot of the average over time of the corresponding time-frequency surface.

The data set seems to be composed of several stationary signals embedded in a noisy background. As in the previous section, the Spectrogram does have narrower spectral ridges than both the IPSLOFAR and ONELOFAR time-frequency surfaces. On closer examination of the surfaces, note the spectral ridge at approximately frequency location 35 in Figure 4.6. The ridge is also seen in the IPSLOFAR surface but is missing altogether in Figure 4.4, the Spectrogram surface. The 1 ½ D surface does clearly locate a frequency component at location 35, but apparently at the expense of obscuring stronger signals in a very 'busy' surface.

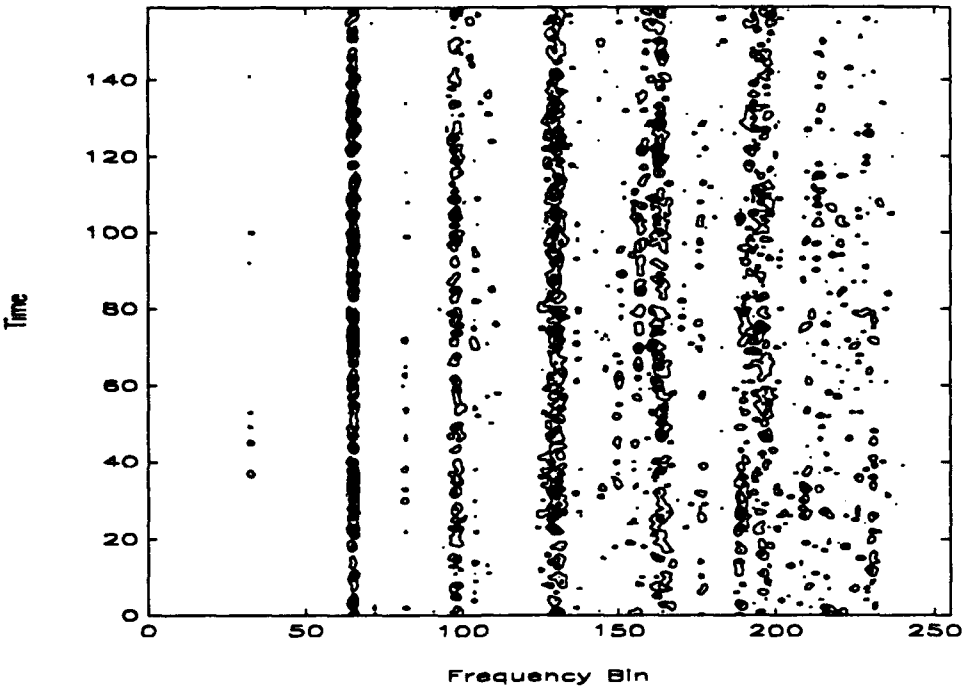


(a)

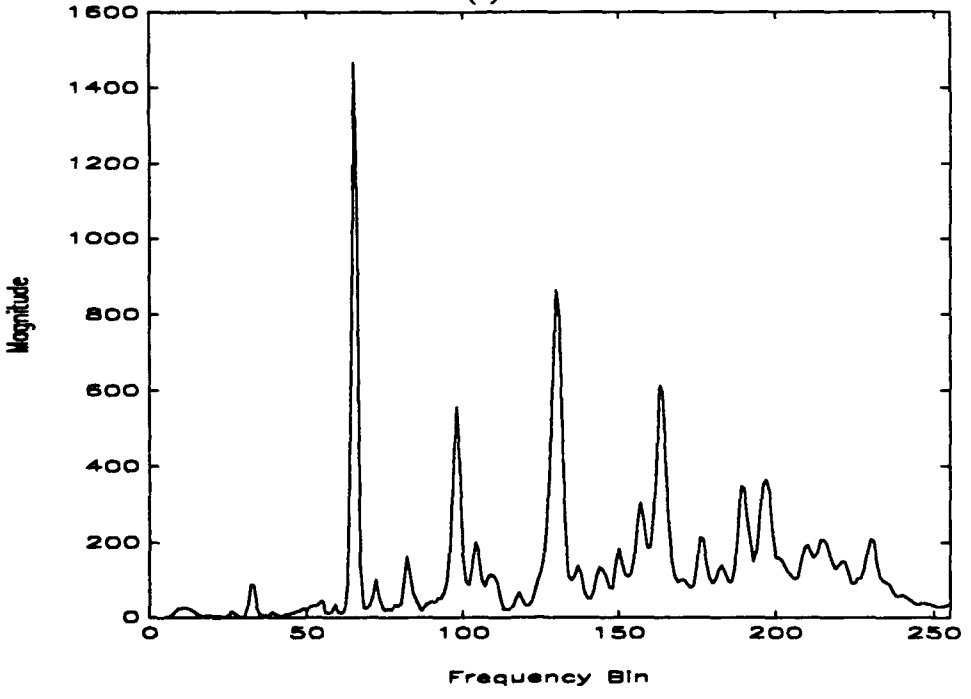


(b)

Figure 4.4 Spectrogram of Acoustic Data

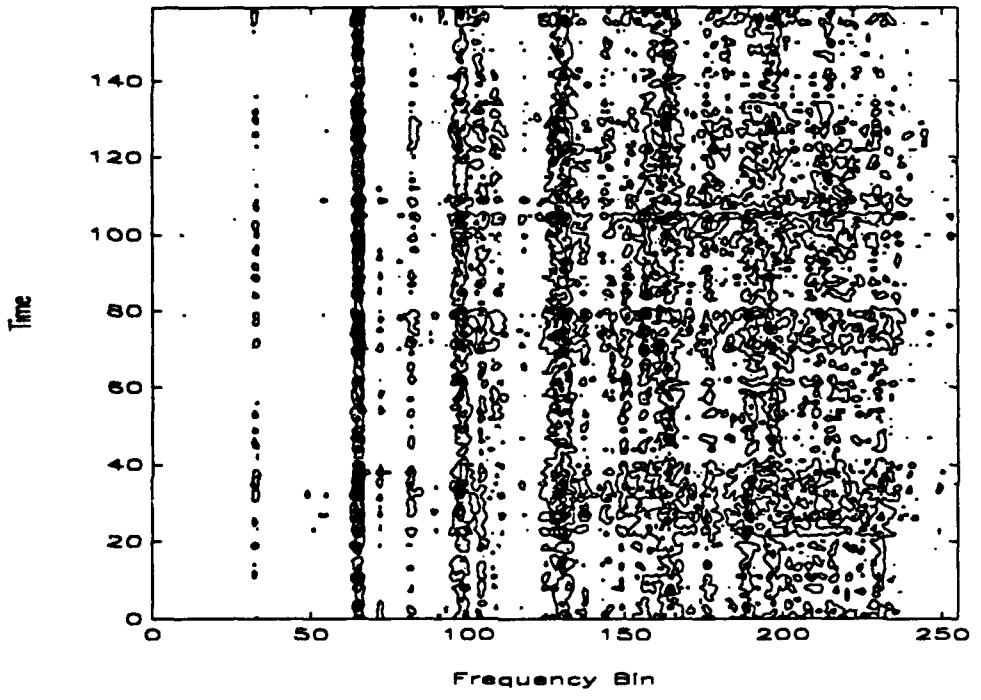


(a)

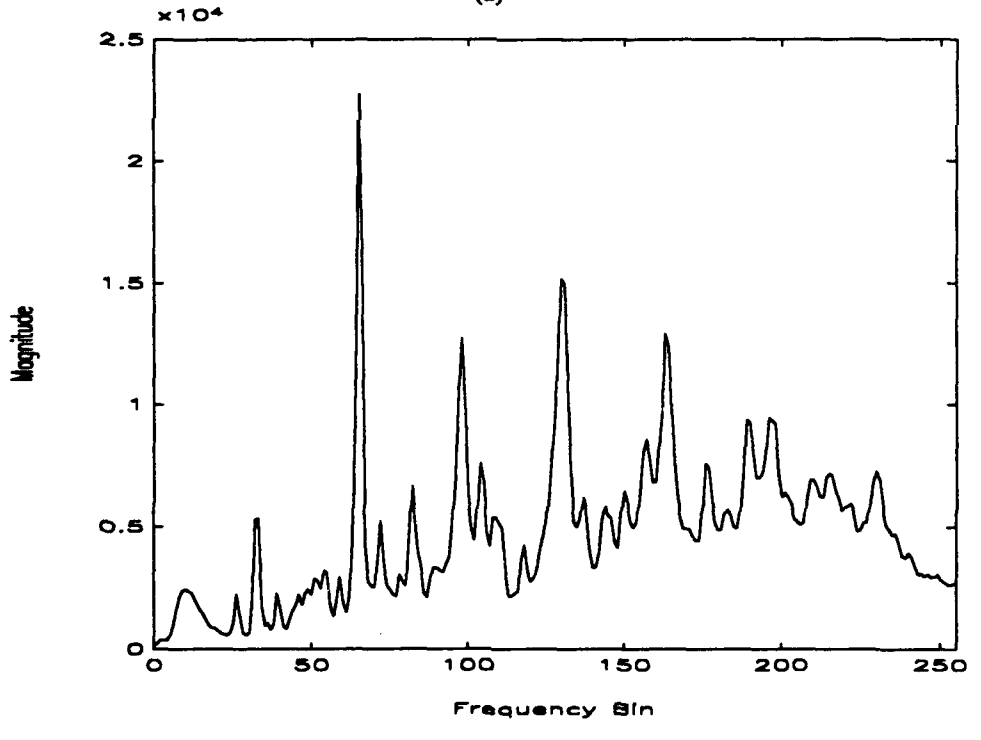


(b)

Figure 4.5. Acoustic Data via IPSLOFAR



(a)



(b)

Figure 4.6. Acoustic Data via ONELOFAR

D. CONCLUSIONS AND SUMMARY

The Spectrogram is well-suited to quickly and clearly resolve stationary signals in moderately noisy environments. The IPS technique is superior to the Spectrogram both in the detection of stationary signals in noise and particularly in the detection of dynamic signals i.e., the Quadratic FM Chirp. Both the IPS and 1 ½ D techniques cannot resolve stationary spectral components as finely as the Spectrogram, but it could be argued that their superior ability to locate spectral components in noisier environments is more valuable in many cases. Potentially, one could also increase the integration time of the IPS based techniques to obtain better resolution. The judicious combination of these techniques can optimize the process of spectral analysis. The Spectrogram can be used as an efficient, first look at a particular data set. Areas of interest, which may indicate dynamic signals could be further analyzed by the IPS technique and finally the 1 ½ D technique can offer superior location of spectral components in noisier environments.

E. AREAS FOR FURTHER RESEARCH

The derivation for the 1 ½ D spectral technique used in this thesis may be further explored to extend its sensitivity to spectral components embedded in a noisy background. The bispectrum surface could be exploited by other than the 1 ½ D technique, which uses a degenerate form of the tricorrelation function. Applications of the Radon transform might shed new insights into the detection and classification of stationary and transient spectral components. Interpretation of each of these techniques depends on the display techniques which are available. The lofar-type displays for the IPSLOFAR and ONELOFAR programs could be enhanced if a true intensity plot could be obtained.

APPENDIX
PROGRAM LISTINGS

```
%function [P,freqindex,timeindex]=ips(data,wintype,winlen,step);
%This function will calculate an Instantaneous Power Spectral (IPS) surface.
%The IPS surface (output matrix P) characteristics are determined by the selection
%of window type (wintype), window length (winlen) and the distance that the
%window is moved through the data sequence (step).
%The program plots only the positive half of the spectral plane.  The
%outputs timeindex and freqindex are the appropriate plot indices for the
%output time-frequency surface.
%
%The inputs are:
%data:      The input data string
%wintype:   '0' Rectangular Window
%          '1' Hamming Window
%winlen:    The desired width of the window, normally half of the siglen
%step:      Time step desired, normally '1' or a multiple of '2'
%
%The outputs are:
%P          The IPS time-frequency surface
%timeindex  The y axis index
%freqindex  The x axis index
%See also IPSSURF, IPSLOFAR

%Karen A. Hagerman
%06 May 1992

function [P,freqindex,timeindex]=ips(data,wintype,winlen,step)

[m,n]=size(data);

if m~=1
  data=data.';
end

siglen=length(data);
if wintype==0
  win=ones(winlen-1,1);
  elseif wintype==1
  win=hamming(winlen-1);
end
W=[win(winlen/2:-1:1)];
x=[zeros(1,winlen) data zeros(1:winlen)];
p=zeros(siglen/step,winlen);
```

```
index=1;
for n=winlen+1:step:siglen+winlen-step+1
  Xm=[conj((x(n:-1:n-(winlen/2-1)))).' (x(n:n+(winlen/2-1))).'];
  Xn=[x(n);conj(x(n))];
  product=((Xm*Xn).*W).';
  product=[product 0 conj(product(winlen/2:-1:2))];
  p(index,:)=fftshift(real(.5*fft(product)));
  index=index+1;
end

p=p(:,winlen/2+1:winlen);
[prow,pcolumn]=size(p);

%Smoothing
p_temp(1,:)=mean(p(1:3,:));
p_temp(2,:)=mean(p(1:4,:));
p_temp(prow-1,:)=mean(p(prow-3:prow,:));
p_temp(prow,:)=mean(p(prow-2:prow,:));
for m=3:prow-2
  p_temp(m,:)=mean(p(m-2:m+2,:));
end

P=flipud(p_temp);

freqindex=[0:pcolumn-1];
timeindex=[1:prow];
```

```
%function [Q,xindex,yindex]=ipssurf(sig,siglen,wintype,winlen,step);
%This function will calculate an Instantaneous Power Spectral (IPS) surface made of
%smaller IPS surfaces. The smaller IPS surfaces are calculated for data sequence pieces
%of length (siglen) of the total input data (sig). Characteristics of the smaller surfaces
%are determined by the selection of window type (wintype), window length (winlen) and
%the distance that the window is moved through the data sequence (step). The surfaces
%are then appended edge to edge to form the output Q surface.
%
%The inputs are:
%sig:       The input data string
%siglen:    The desired length of the discrete pieces of sig
%wintype:   '0' Rectangular Window
%          '1' Hamming Window
%winlen:    The desired width of the window, normally half of the siglen
%step:      Time step desired, normally '1' or a multiple of '2'
%
%The outputs are:
%P          The IPSSURF time-frequency surface
%yindex     The y axis index
%xindex     The x axis index
%
%See also IPS, IPSLOFAR

%Karen A. Hagerman
%06 May 1992

function [Q,xindex,yindex]=ipssurf(data,siglen,wintype,winlen,step)

if nargin==1
  siglen=input('Enter the desired length of the sequence ');
  wintype=input('ENTER "0" for RECT. WINDOW or "1" for HAMMING WINDOW');
  winlen=input('ENTER length of the window (must be an even number):');
  step=input('Input desired step in time ');
end

[m,n]=size(data);

if m~=1
  data=data.';
end

rows=floor(length(data)/siglen);
```

```
data=[zeros(1,winlen) data zeros(1,winlen)];
len=length(data);
finish1=winlen/2-1;
finish2=finish1+1;

if wintype==0
  win=ones(winlen-1,1);
  elseif wintype==1
  win=hamming(winlen-1);
end

W=[win(winlen/2:-1:1)]';
Q=zeros(rows*(siglen/step),winlen/2);
qindex=1:siglen/step:(rows*siglen/step);
l=1;
for k=1:siglen:len-siglen-2*winlen+1
  x=data(1,k:k+siglen+2*winlen-1);
  index=1;
  for n=winlen+1:step:siglen+winlen-step+1
    Xm=[conj((x(n:-1:n-finish1))); (x(n:n+finish1))];
    Xn=[x(n) conj(x(n))];
    product=((Xn*Xm).*W);
    product=[product 0 conj(product(finish2:-1:2))];
    P(index,:)=(fftshift(real(.5*fft(product))));
    index=index+1;
  end
  Q(qindex(l):qindex(l)+(siglen/step)-1,:)=P(:,winlen/2+1:winlen);
  l=l+1;
end

Q=flipud(Q);
[qrow,qcolumn]=size(Q);
xindex=[0:qcolumn-1];
yindex=[1:qrow];
```

```
%function[Q,freqindex,timeindex]=ipslofar(data,siglen,wintype,winlen,step);
%This function will calculate a 'Lofar' display for a selected data sequence.
%The total data sequence is first divided into equal length pieces of
%length (siglen). An IPS surface is then calculated for each piece. The
%IPS surface characteristics are determined by the selection of window
%type (wintype), window length (winlen) and the distance that the window
%is moved through the data sequence pieces (step). The mean is then taken
%of the IPS surface and is placed sequentially in the Q matrix for display.
%The Q matrix plots only the positive half of the spectral plane. The
%outputs timeindex and freqindex can be used in plots to interpret the
%results.
%
%The inputs are:
%data:      The input data string
%siglen:    The desired length of the discrete parts of the sequence
%wintype:   '0' Rectangular Window
%          '1' Hamming Window
%winlen:    The desired width of the window, normally half of the siglen
%step:      Time step desired, normally '1' or a multiple of '2'
%
%The outputs are:
%P          The IPSLOFAR time-frequency surface
%timeindex  The y axis index
%freqindex  The x axis index
%
%See also IPS, IPSSURF

%Karen A. Hagerman
%06 May 1992

function [Q,freqindex,timeindex]=ipslofar(data,siglen,wintype,winlen,step)

if nargin==1
  siglen=input('Enter the desired length of the sequence ');
  wintype=input('ENTER "0" for RECT. WINDOW or "1" for HAMMING WINDOW:');
  winlen=input('ENTER length of the window (must be an even number):  ');
  step=input('Input desired step in time ');
end
[m,n]=size(data);
if m~=1
  data=data.';
end
rows=floor(length(data)/siglen);
```

```
Q=zeros(rows,winlen/2);
data=[zeros(1,winlen) data zeros(1,winlen)];
len=length(data);
finish1=winlen/2-1;
finish2=finish1+1;
if wintype==0
  win=ones(winlen-1,1);
  elseif wintype==1
  win=hamming(winlen-1);
end
W=[win(winlen/2:-1:1)]';

qindex=1;
for k=1:siglen:len-siglen-2*winlen+1
  x=data(1,k:k+siglen+2*winlen-1);
  index=1;
  for n=winlen+1:step:siglen+winlen-step+1
    Xm=[conj((x(n:-1:n-finish1))); (x(n:n+finish1))];
    Xn=[x(n) conj(x(n))];
    product=((Xn*Xm).*W);
    product=[product 0 conj(product(finish2:-1:2))];
    P(index,:)=(fftshift(real(.5*fft(product))));
    index=index+1;
  end
  if index ~=2
    Q(qindex,:)=mean(P(:,winlen/2+1:winlen));
    qindex=qindex+1;
  else
    Q(qindex,:)=P(1,winlen/2+1:winlen);
    qindex=qindex+1;
  end
end
Q=flipud(Q);

[qrow,qcolumn]=size(Q);
freqindex=[0:qcolumn-1];
timeindex=[1:qrow];
```

```
%function [P,freqindex,timeindex]=one_half(data,wintype,winlen,step);
%This function will calculate the 1 1/2 D Spectral surface.  The 1 1/2
%D surface characteristics are determined by the selection of window
%type (wintype), window length (winlen) and the distance that the window
%is moved through the data sequence (step).
%The 1 ½ D surface (output matrix P) characteristics are determined by the selection
%of window type (wintype), window length (winlen) and the distance that the
%window is moved through the data sequence (step).
%The program plots only the positive half of the spectral plane.  The
%outputs timeindex and freqindex are the appropriate plot indices for the
%output time-frequency surface.
%
%The inputs are:
%data:      The input observations vector, for maximum effectiveness should
%          be of a length which is a power of 2, e.g. 64,128,512
%wintype:   '0' Rectangular Window
%          '1' Hamming Window
%winlen:    The desired width of the window, normally half of the input
%          length
%step:      Time step desired, can be '1' or a multiple of '2'
%
%The outputs are:
%P          The 1 ½ D time-frequency surface
%timeindex  The y axis index
%freqindex  The x axis index
%
%See also ONESURF, ONELOFAR

%Karen A. Hagerman
%06 May 1992

function [P,freqindex,timeindex]=one_half(data,wintype,winlen,step)

[datarows,datacolumns]=size(data);
if datarows ~=1
  data=data.';
end

siglen=length(data);

if wintype==0
  win=ones(winlen-1,1);
  elseif wintype==1
```

```
  win=hamming(winlen-1);
end

W=win(winlen/2:-1:1);
x=[zeros(1,winlen) data zeros(1:winlen)].';
p=zeros(siglen/step,winlen);

index=1;
for n=winlen+1:step:siglen+winlen-step+1
  Xn=[abs(x(n))^2; abs(x(n))^2];
  Xm=[conj(x(n:-1:n-(winlen/2-1))) x(n:n+(winlen/2-1))];
  product=((Xm*Xn).*W).';
  product=[product 0 conj(product(winlen/2:-1:2))];
  p(index,:)=fftshift(abs(.5*fft(product)));
  index=index+1;
end
p=p(:,winlen/2+1:winlen);
[prow,pcolumn]=size(p);

%Smoothing
p_temp(1,:)=mean(p(1:3,:));
p_temp(2,:)=mean(p(1:4,:));
p_temp(prow-1,:)=mean(p(prow-3:prow,:));
p_temp(prow,:)=mean(p(prow-2:prow,:));
for m=3:prow-2
   p_temp(m,:)=mean(p(m-2:m+2,:));
end

P=flipud(p_temp);

freqindex=[0:pcolumn-1];
timeindex=[1:prow];
```

```
%function [Q,xindex,yindex]=onesurf(sig,siglen,wintype,winlen,step);
%This function will calculate an 1 1/2 D spectral surface made of
%smaller 1 1/2 D surfaces. The smaller IPS surfaces are calculated for
%data sequence pieces of length (siglen) of the total input data (sig).
%Characteristics of the smaller surfaces are determined by the selection of
%window type (wintype), window length (winlen) and the distance that the
%window is moved through the data sequence (step). The surfaces are then
%appended edge to edge to form the output Q surface.
%
%The inputs are:
%sig:       The input observation sequence vector
%siglen:    The desired length of the discrete pieces of the input vector
%wintype:   '0' Rectangular Window
%          '1' Hamming Window
%winlen:    The desired width of the window, normally half of the input
%          vector length
%step:      Time step desired, normally '1' or a multiple of '2'
%
%The outputs are:
%P          The ONESURF time-frequency surface
%yindex     The y axis index
%xindex     The x axis index
%
%See also ONE_HALF, ONELOFAR

%Karen A. Hagerman
%06 May 1992

function [Q,xindex,yindex]=onesurf(data,siglen,wintype,winlen,step)

if nargin==1
  siglen=input('Enter the desired length of the sequence ');
  wintype=input('ENTER "0" for RECT. WINDOW or "1" for HAMMING WINDOW:');
  winlen=input('ENTER length of the window (must be an even number):  ');
  step=input('Input desired step in time ');
end
[m,n]=size(data);
if m~=1
  data=data.';
end
rows=floor(length(data)/siglen);
data=[zeros(1,winlen) data zeros(1,winlen)];
len=length(data);
```

```
finish1=winlen/2-1;
finish2=finish1+1;

if wintype==0
  win=ones(winlen-1,1);
  elseif wintype==1
  win=hamming(winlen-1);
end
W=[win(winlen/2:-1:1)]';
Q=zeros(rows*(siglen/step),winlen/2);
qindex=1:siglen/step:(rows*siglen/step);
l=1;
for k=1:siglen:len-siglen-2*winlen+1
  x=data(1,k:k+siglen+2*winlen-1);
  index=1;
  for n=winlen+1:step:siglen+winlen-step+1
    Xn=[abs(x(n))^2 abs(x(n))^2];
    Xm=[conj(x(n:-1:n-(winlen/2-1))); x(n:n+(winlen/2-1))];
    product=((Xn*Xm).*W);
    product=[product 0 conj(product(winlen/2:-1:2))];
    P(index,:)=fftshift(real(.5*fft(product)));
    index=index+1;
  end
  Q(qindex(l):qindex(l)+(siglen/step)-1,:)=P(:,winlen/2+1:winlen);
  l=l+1;
end
Q=flipud(abs(Q));
[qrow,qcolumn]=size(Q);
xindex=[0:qcolumn-1];
yindex=[1:qrow];
```

```
%function [Q,freqindex,timeindex]=onelofar(sig,siglen,wintype,winlen,step);
%This function will calculate a 'Lofar' display for a selected data sequence.
%The total data sequence is first divided into equal length pieces of
%length (siglen). An 1 1/2 D surface is then calculated for each piece. The
%1 1/2 D surface characteristics are determined by the selection of window
%type (wintype), window length (winlen) and the distance that the window
%is moved through the data sequence pieces (step). The mean is then taken
%of the 1 1/2 D surface and is placed sequentially in the Q matrix for display.
%The Q matrix plots only the positive half of the spectral plane. The
%outputs timeindex and freqindex can be used in plots to interpret the
%results.
%
%The inputs are:
%data:      The input data string
%siglen:    The desired length of the discrete parts of the sequence
%wintype:   '0' Rectangular Window
%          '1' Hamming Window
%winlen:    The desired width of the window, normally half of the siglen
%step:      Time step desired, normally '1' or a multiple of '2'
%See also ONE_HALF, ONESURF
%
%The outputs are:
%P          The ONELOFAR time-frequency surface
%timeindex  The y axis index
%freqindex  The x axis index
%

%Karen A. Hagerman
%06 May 1992

function [Q,freqindex,timeindex]=onelofar(sig,siglen,wintype,winlen,step)

if nargin==1
  siglen=input('Enter the desired length of the sequence ');
  wintype=input('ENTER "0" for RECT. WINDOW or "1" for HAMMING WINDOW:');
  winlen=input('ENTER length of the window (must be an even number):  ');
  step=input('Input desired step in time ');
end

[m,n]=size(sig);
if m~=1
  sig=sig.';
end
```

```
rows=floor(length(sig)/siglen);
Q=zeros(rows,winlen/2);
sig=[zeros(1,winlen) sig zeros(1,winlen)];
len=length(sig);

if wintype==0
  win=ones(winlen-1,1);
  elseif wintype==1
  win=hamming(winlen-1);
end
W=win(winlen/2:-1:1).';

qindex=1;
for k=1:siglen:len-siglen-2*winlen+1
  x=sig(1,k:k+siglen+(2*winlen)-1);
  index=1;
  for l=winlen+1:step:siglen+winlen-step+1
    Xn=[abs(x(l))^2 abs(x(l))^2];
    Xm=[conj(x(l:-1:l-(winlen/2-1))); x(l:l+(winlen/2-1))];
    product=((Xn*Xm).*W);
    product=[product 0 conj(product(winlen/2:-1:2))];
    p(index,:)=fftshift(abs(.5*fft(product)));
    index=index+1;
  end
  if index ~=2
    Q(qindex,:)=mean(p(:,winlen/2+1:winlen));
    qindex=qindex+1;
  else
    Q(qindex,:)=p(1,winlen/2+1:winlen);
    qindex=qindex+1;
  end
end
Q=flipud(Q);

[qrow,qcolumn]=size(Q);
freqindex=[0:qcolumn-1];
timeindex=[1:qrow];
```

References

1. Lederman, Leon M. and Schramm, David N., *From Quarks to the Cosmos, Tools of Discovery,* Scientific American Library, 1989.

2. Kay, Steven M., *Modern Spectral Estimation,* Prentice-Hall, 1988.

3. Therrien, Charles W., *Discrete Random Signals and Statistical Signal Processing,* Prentice-Hall, 1992.

4. Stitz, Elizabeth H., *Instantaneous Power Spectrum*, Master's Thesis, Naval Postgraduate School, Monterey, California, 1990.

5. Monica de Oliveira, Paulo M. D., *Instantaneous Power Spectrum,* Master's Thesis, Naval Postgraduate School, Monterey, California, 1989.

6. Mendel, Jerry M., "Tutorial on Higher-Order Statistics (Spectra) in Signal Processing and System Theory: Theoretical Results and Some Applications," *Proc. IEEE,* v. 79, pp. 278-305, 1991.

7. Kay, Steven M. and Marple JR., Stanley Lawrence, "Spectrum Analysis-A Modern Perspective," *proc. IEEE,* v. 69, pp. 1380-1489, 1981.

8. Page, C. H., "Instantaneous Power Spectra," *Journal of Applied Physics,* v. 23, pp. 103-206, January 1952.

9. Levin, M. J., "Instantaneous Spectra and Ambiguity Functions," *IEEE Transactions Information Theory,* v. IT-10, pp. 95-97, January 1964.

INITIAL DISTRIBUTION LIST

1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22304-6145	2
2.	Library, Code 52 Naval Postgraduate School Monterey, California 93943-5002	2
3.	Chairman, Code EC Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, California 93943-5000	1
4.	Prof. R. Hippenstiel, Code EC/Hi Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, California 93943-5000	2
5.	Prof. M.P. Fargues, Code EC/Fa Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, California 93943-5000	1
6.	LT Karen A. Hagerman PSC 818, Box 115 FPO AP 09644-2000	1
7.	Naval Ocean Systems Center ATTN: Dr. C.E. Persons (Code 732) San Diego, California 92152	1